Computational Assemblies: Analysis, Design, and Fabrication

[1] Peng Song [2] Ziqi Wang [3] Marco Livesu

[1] Singapore University of Technology and Design, Singapore
[2] Swiss Federal Institute of Technology Zürich, Switzerland
[3] Institute of Applied Mathematics and Information Technologies, CNR, Italy

Presenters

Peng Song

Ziqi Wang

Marco Livesu

What is this tutorial about?

- **Introduce** computational techniques to **analyze**, **design**, and **fabricate** assemblies, an emerging research area named **computational assemblies**
- **Review and categorize** recent research works on computational assemblies, focusing on representations, methods, and algorithms
- Provide **insight** of challenges and future research directions
- Target audience
  - Students and researchers
  - Designers and makers

Timetable

		Peng	Ziqi	Marco
Introduction	~20 mins	X		
Computational analysis of assemblies	~50 mins	X		
Computational design of assemblies	~50 mins		X	
Computational fabrication of assemblies	~50 mins			X
Q & A	~10 mins	X	X	X

Timetable

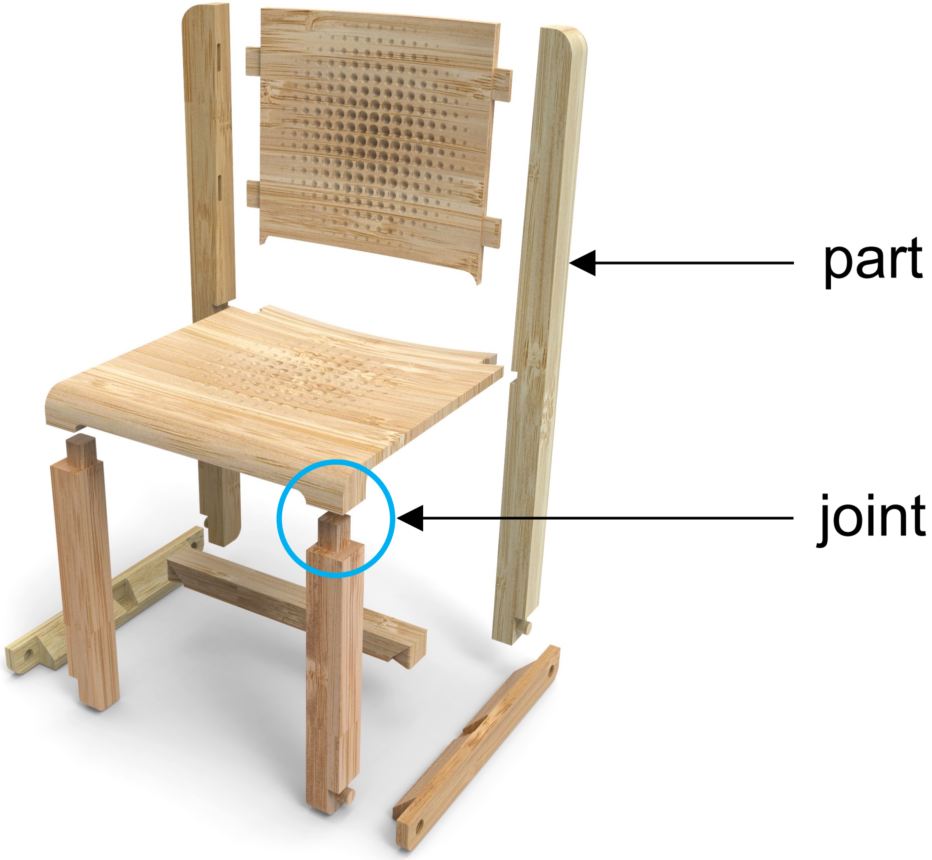
		Peng	Ziqi	Marco
Introduction	~20 mins	X		
Computational analysis of assemblies	~50 mins	X		
Computational design of assemblies	~50 mins		X	
Computational fabrication of assemblies	~50 mins			X
Q & A	~10 mins	X	X	X

Assemblies

An assembly is an arrangement of parts connected by joints to have a specific form and functionality.

Assemblies

An assembly is an arrangement of parts connected by joints to have a specific form and functionality.



Assemblies

Monolithic object

VS

Assembly

Assemblies: Advantages

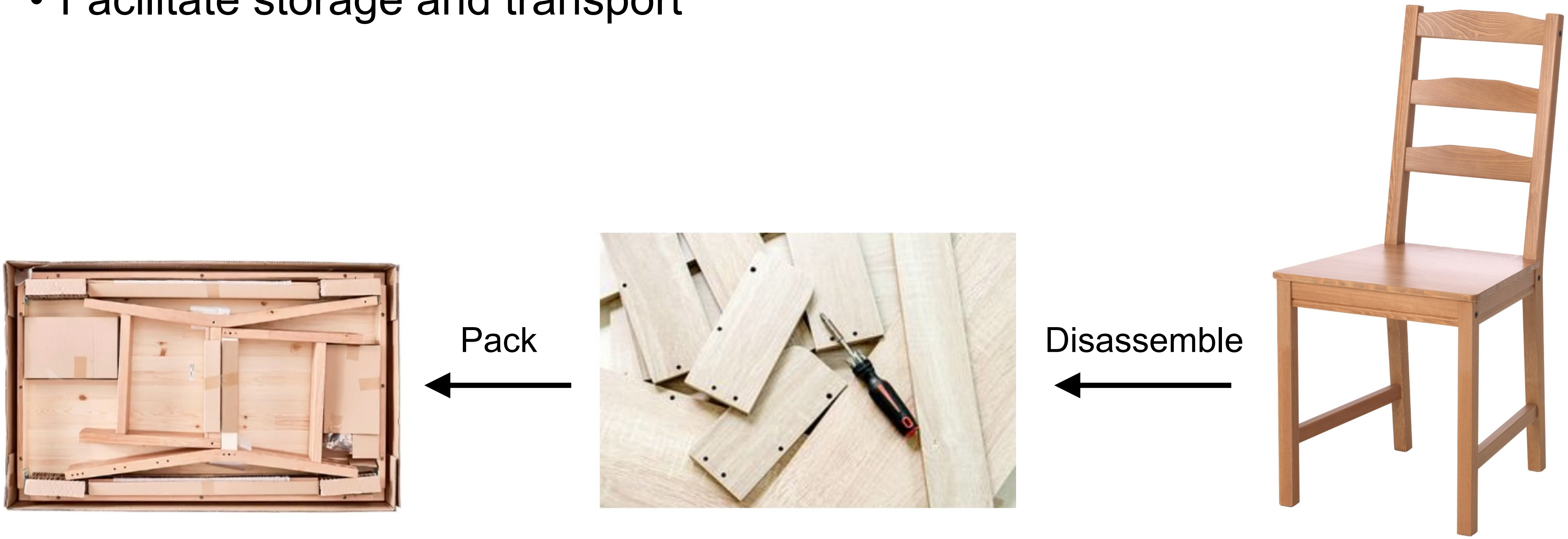
- Simplify fabrication

Fabricate

Assemble

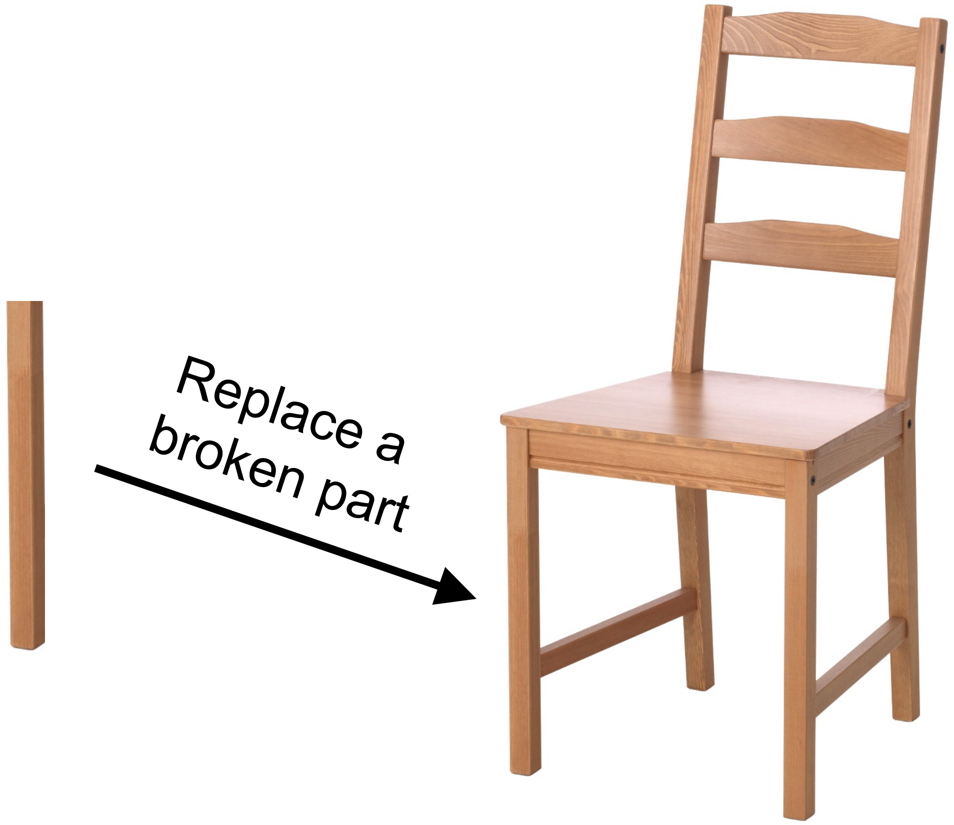
Assemblies: Advantages

- Simplify fabrication
- Facilitate storage and transport



Assemblies: Advantages

- Simplify fabrication
- Facilitate storage and transport
- Facilitate maintenance



Assemblies: Advantages

- Simplify fabrication
- Facilitate storage and transport
- Facilitate maintenance
- Multiple forms

Assemblies: Advantages

- Simplify fabrication
- Facilitate storage and transport
- Facilitate maintenance
- Multiple forms
- Multiple functionalities

Designing Assemblies for Functionality

- Designing assemblies for use in different applications

Toy

Device

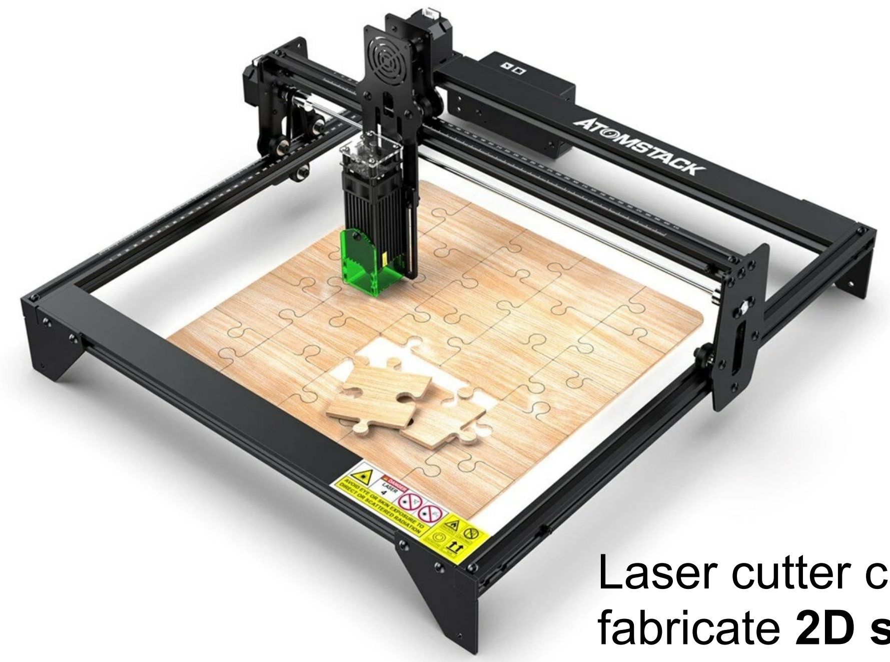
Furniture

Architecture

Designing Assemblies for Form

- Designing assemblies for fabricating complex 3D shapes

How can we fabricate a 3D bunny using 2D laser cutting?

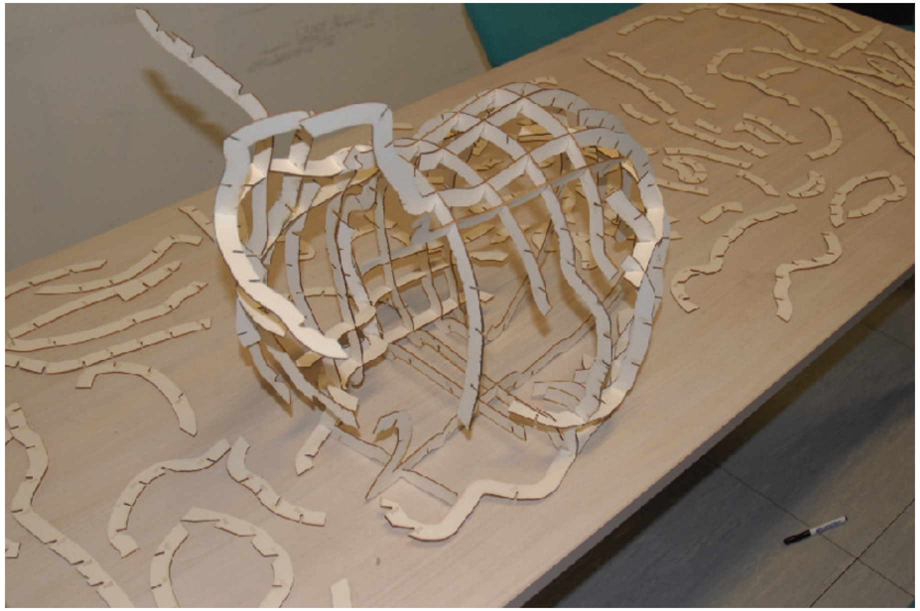


Laser cutter can only fabricate **2D shapes**.

Designing Assemblies for Form

- Designing assemblies for fabricating complex 3D shapes

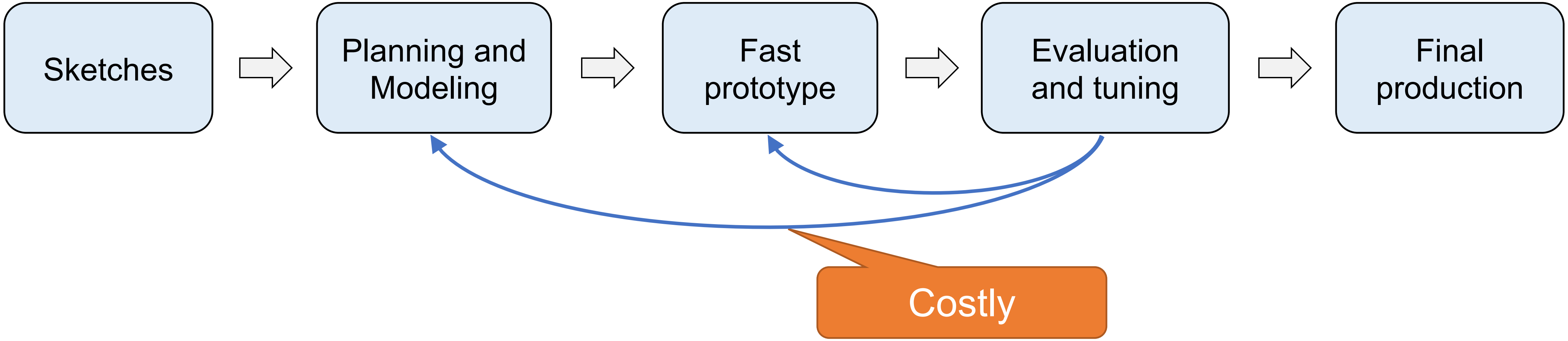
Solution: fabricate a 3D bunny as an assembly with **planar** parts



[Cignoni et al. 2014]

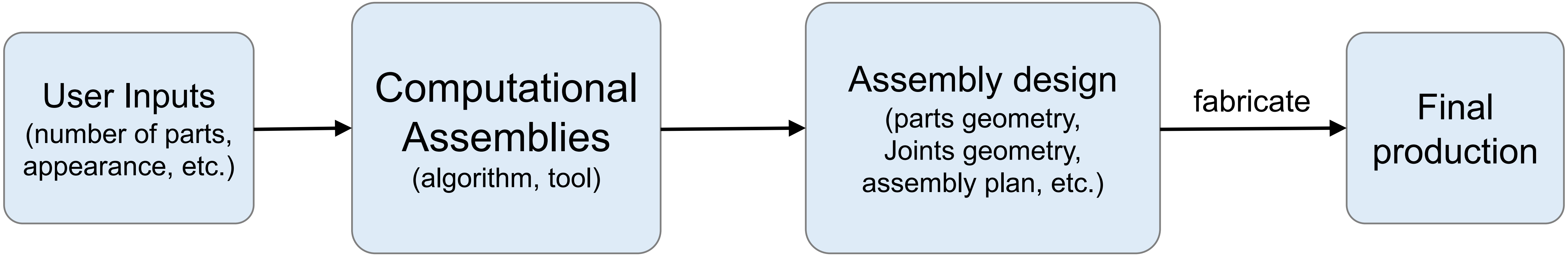
Traditional Design Process

- Traditionally, designing assemblies is a challenging task restricted to the professionals.



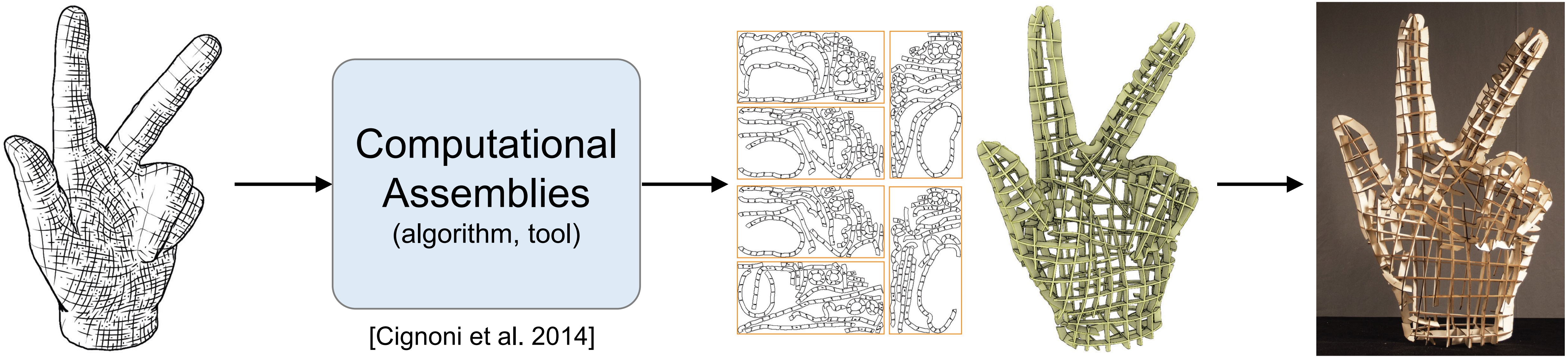
Computational Assemblies

- Currently, there is a trend to study and develop computational techniques for **analyzing**, **designing**, and **fabricating** assemblies
- We name this emerging research area **computational assemblies**
  - enable general users to design **personalized assemblies**
  - enable to generate designs with **optimized performance**



Computational Assemblies

- Currently, there is a trend to study and develop computational techniques for **analyzing**, **designing**, and **fabricating** assemblies
- We name this emerging research area **computational assemblies**
  - enable general users to design **personalized assemblies**
  - enable to generate designs with **optimized performance**

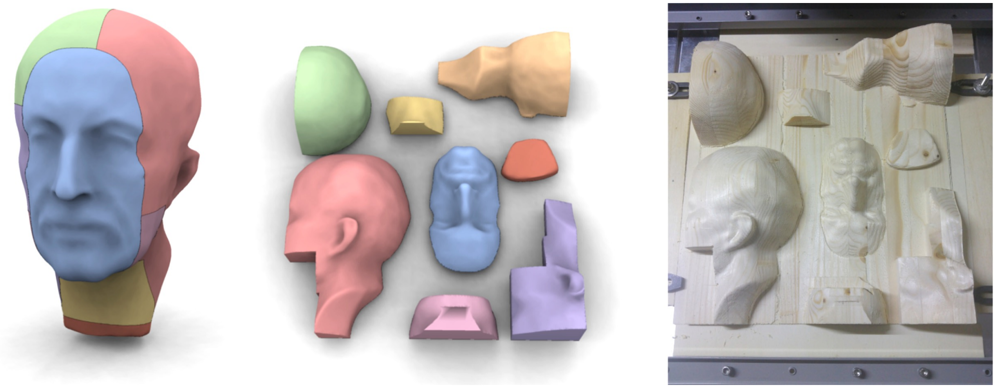


[Cignoni et al. 2014]

Research in Computational Assemblies

- Parts fabricability

CNC milling



[Muntoni et al. 2018]

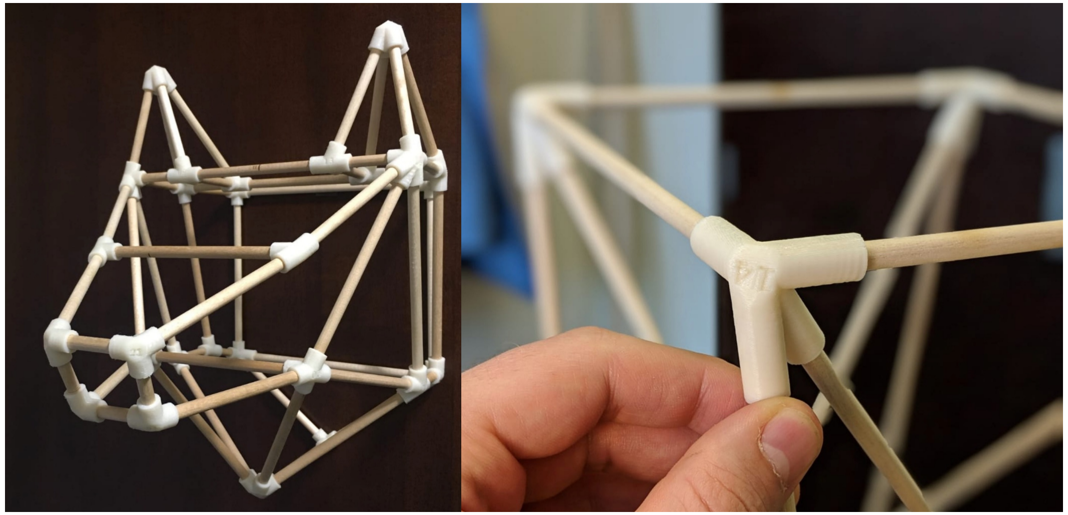
laser cutting
+ 3D printing

[Song et al. 2016]

Research in Computational Assemblies

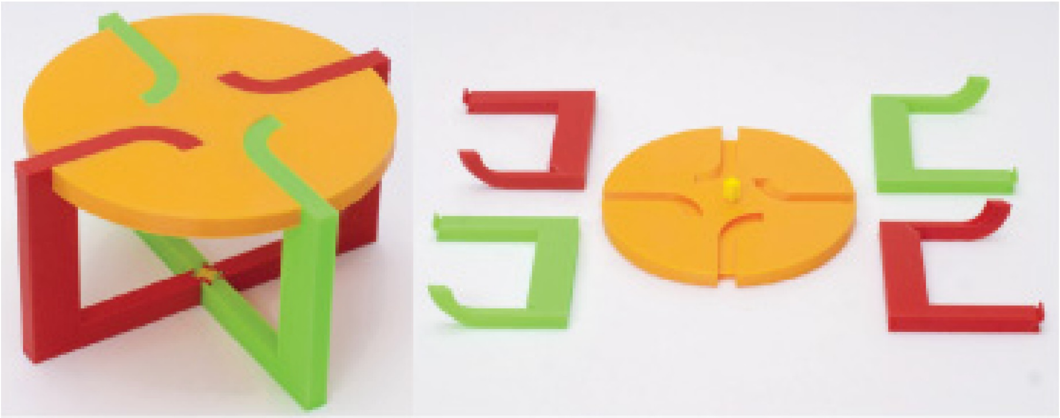
- Parts fabricability
- Parts joining

external joint



[Jacobson 2019]

integral joint

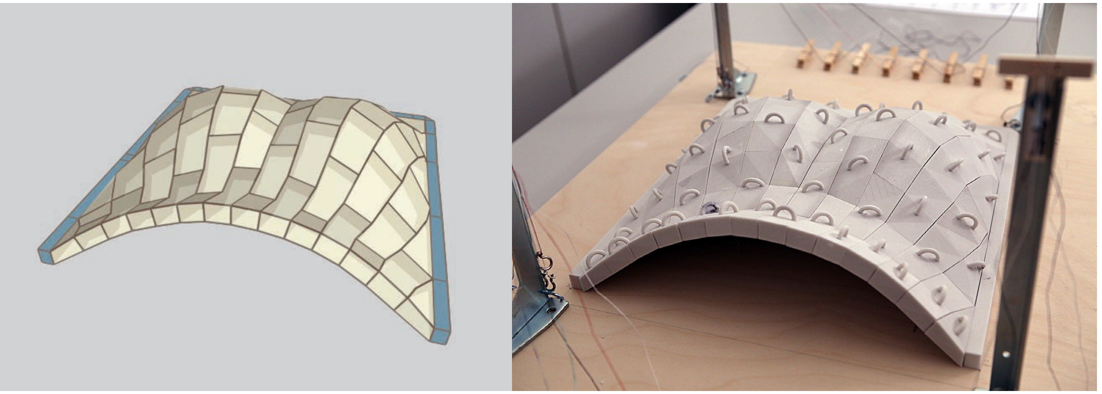


[Yao et al. 2019]

Research in Computational Assemblies

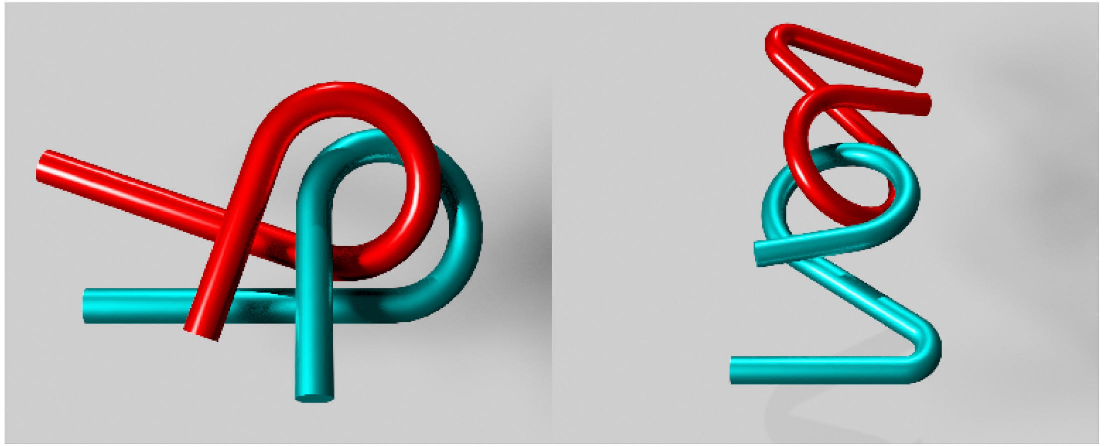
- Parts fabricability
- Parts joining
- Parts assembly

assembly
sequence



[Deuss et al. 2014]

assembly
motion



[Zhang et al. 2020]

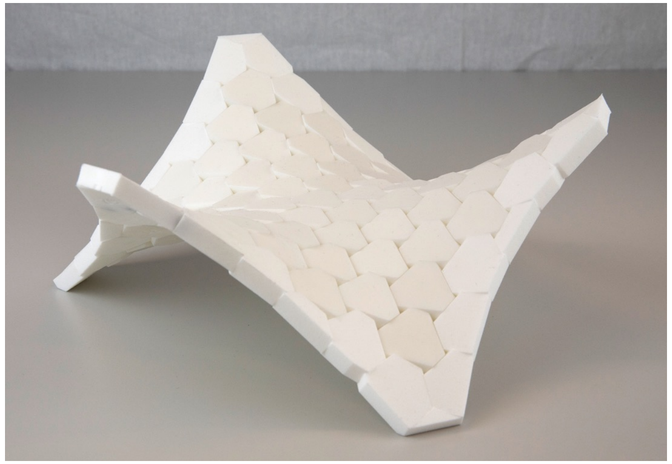
Research in Computational Assemblies

- Parts fabricability
- Parts joining
- Parts assembly
- Structural stability

equilibrium

[Panozzo et al. 2013]

interlocking



[Wang et al. 2019]

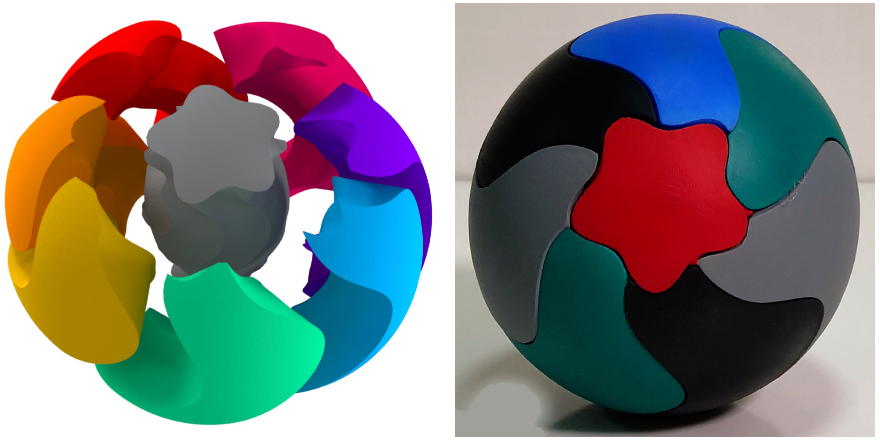
Research in Computational Assemblies

- Parts fabricability
- Parts joining
- Parts assembly
- Structural stability
- Assembly aesthetics

cutting seams

[Filoscia et al. 2020]

multi-color



[Araújo et al. 2019]

Research in Computational Assemblies

- Parts fabricability
- Parts joining
- Parts assembly
- Structural stability
- Assembly aesthetics
- Reconfigurability

free reconfig.

[Song et al. 2017]

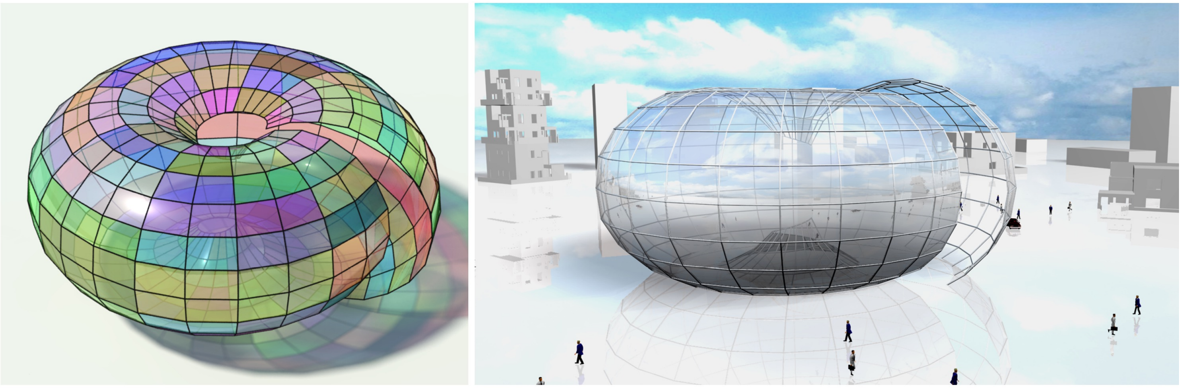
[Yuan et al. 2018]

hinged reconfig.

Research in Computational Assemblies

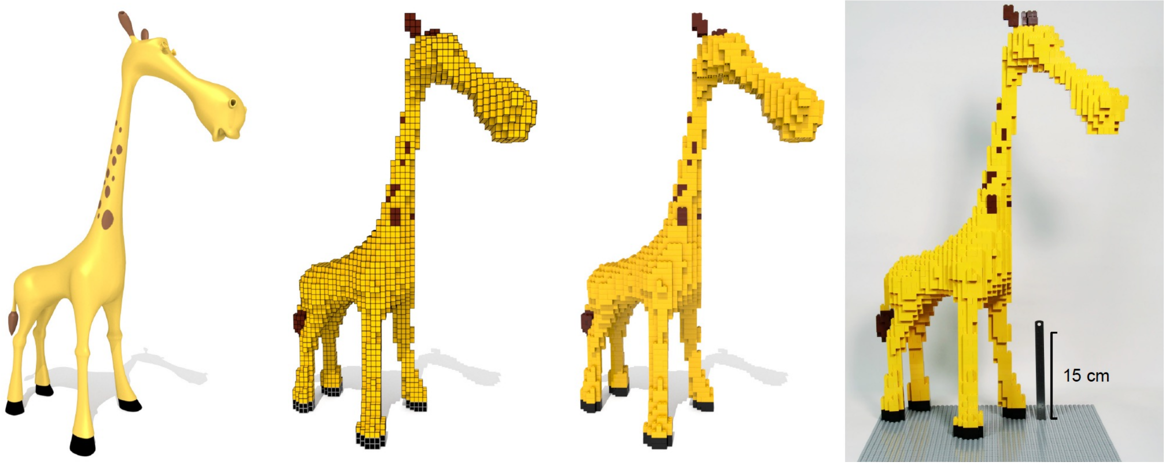
- Parts fabricability
- Parts joining
- Parts assembly
- Structural stability
- Assembly aesthetics
- Reconfigurability
- Reusability

top-down



[Fu et al. 2010]

bottom-up

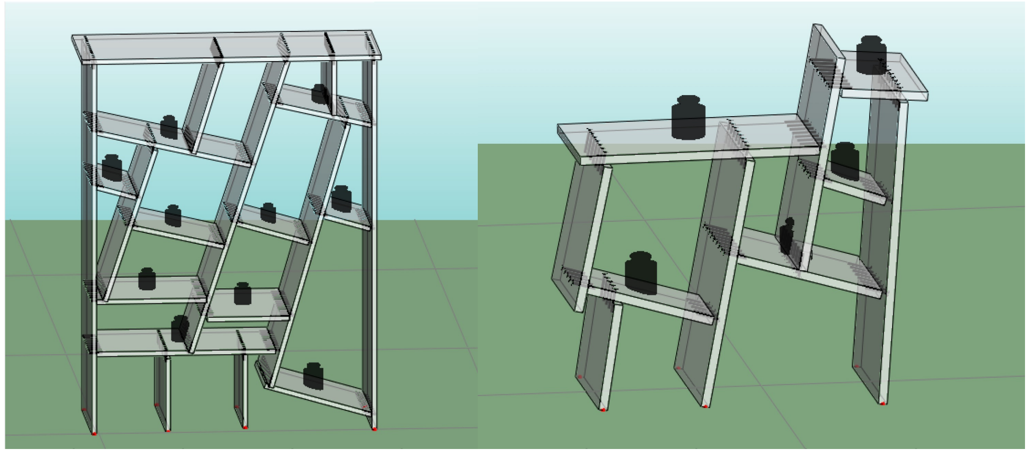


[Luo et al. 2015]

Research in Computational Assemblies

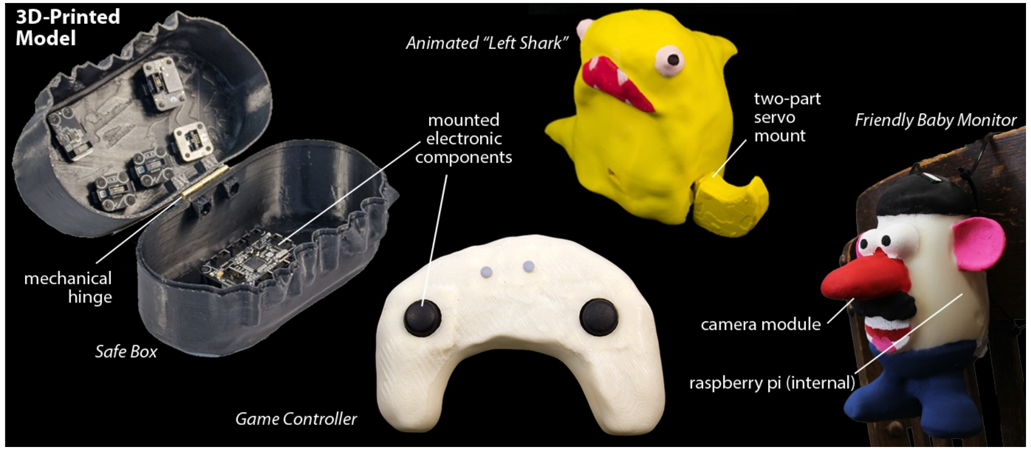
- Parts fabricability
- Parts joining
- Parts assembly
- Structural stability
- Assembly aesthetics
- Reconfigurability
- Reusability
- Functionality

bear force



[Umetani et al. 2012]

interaction



[Savage et al. 2015]

Research in Computational Assemblies

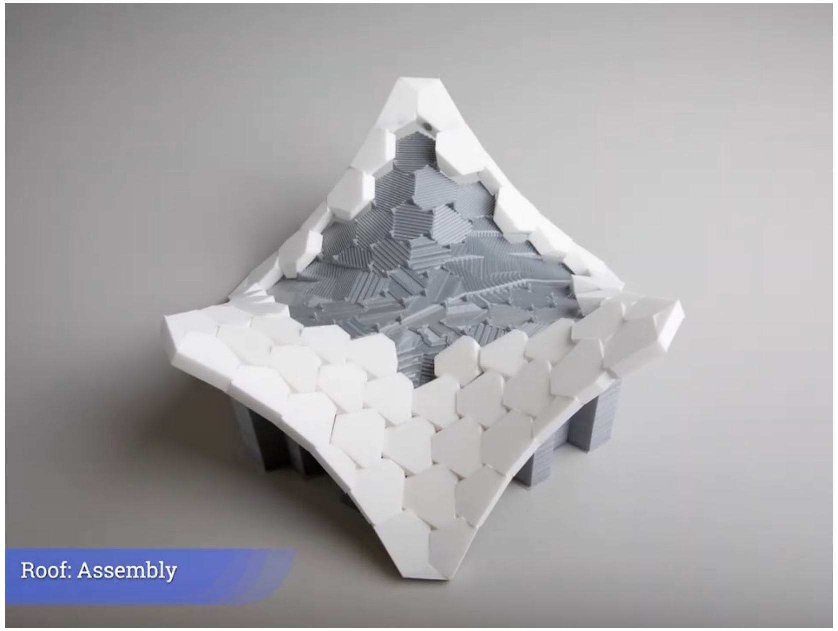
- Computational analysis of assemblies
  - evaluate/quantify designs
  - guide the design process
- Computational design of assemblies
  - design assemblies for use (i.e., functionality)
- Computational fabrication of assemblies
  - design assemblies for fabrication (i.e., form)

Our Scope

Assemblies

Structures

Mechanisms



[Wang et al. 2019]

[Coros et al. 2013]

Our Scope

Assemblies

Structures ✓ | Mechanisms

	Structures	**Mechanisms**
Functionality	transfer force	transfer motion and force
Applications	furniture, architecture, puzzle	machine, robot, mechanical toy
Typical parts	elongated, planar, or solid parts	gear, linkage, cam
Design focus	structural stability	kinematic and dynamic performance

Roof: Assembly

[Wang et al. 2019]

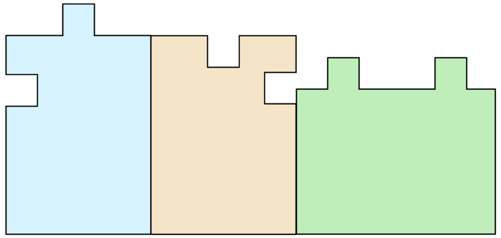
[Coros et al. 2013]

Timetable

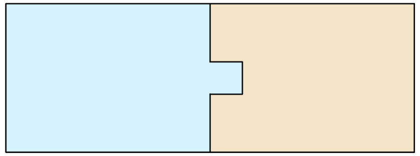
		Peng	Ziqi	Marco
Introduction	~20 mins	X		
Computational analysis of assemblies	~50 mins	X		
Computational design of assemblies	~50 mins		X	
Computational fabrication of assemblies	~50 mins			X
Q & A	~10 mins	X	X	X

Computational Analysis of Assemblies

Parts fabricability

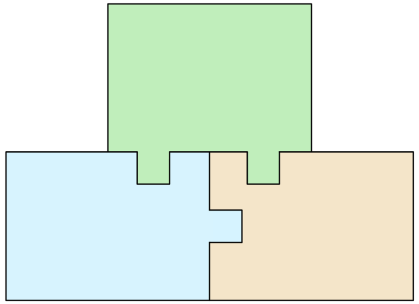


Parts joining



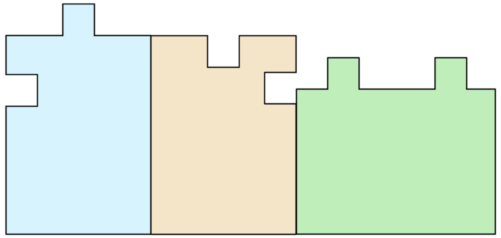
Assembly planning

Structural stability

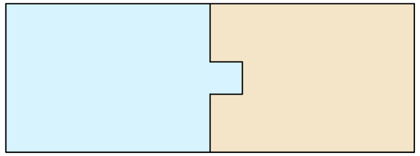


Computational Analysis of Assemblies

Parts fabricability

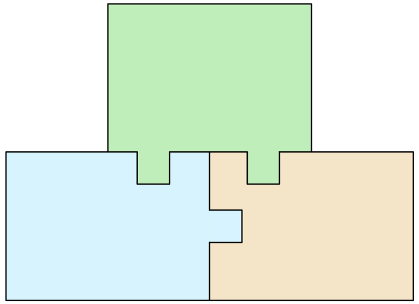


Parts joining



Assembly planning

Structural stability

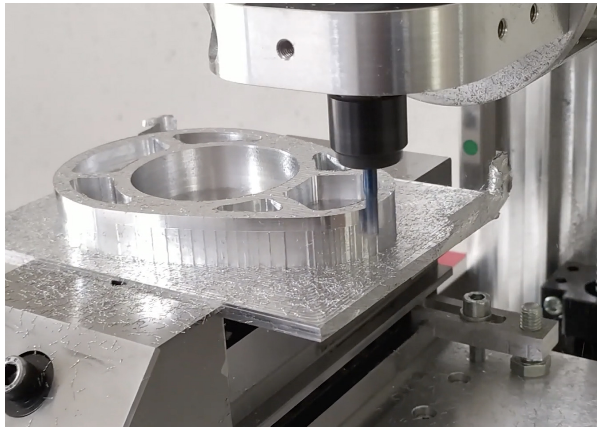


Parts Fabricability

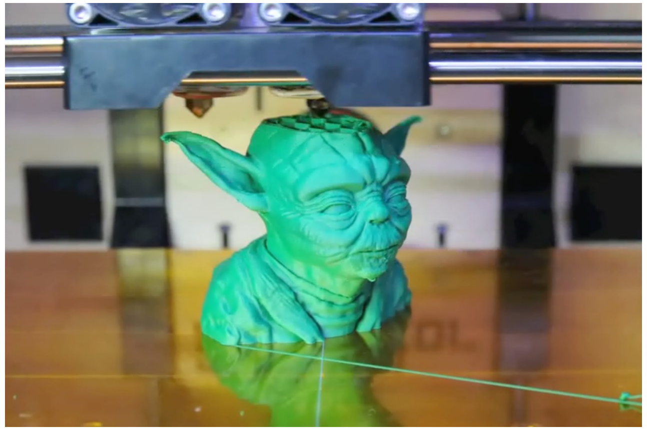
- Parts fabricability depends on the selected **digital fabrication** technique

Laser Cutting

CNC Milling



3D Printing



Subtractive manufacturing

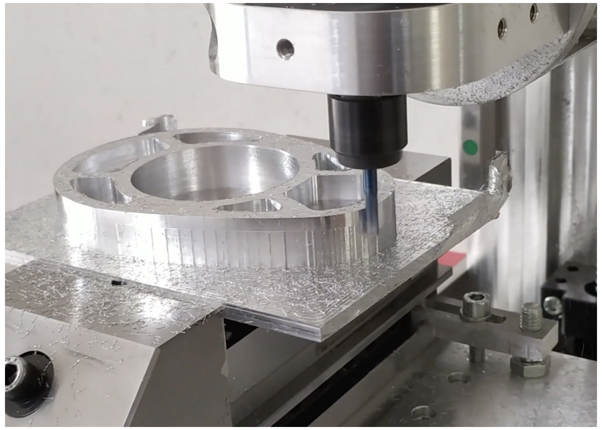
Additive manufacturing

Parts Fabricability

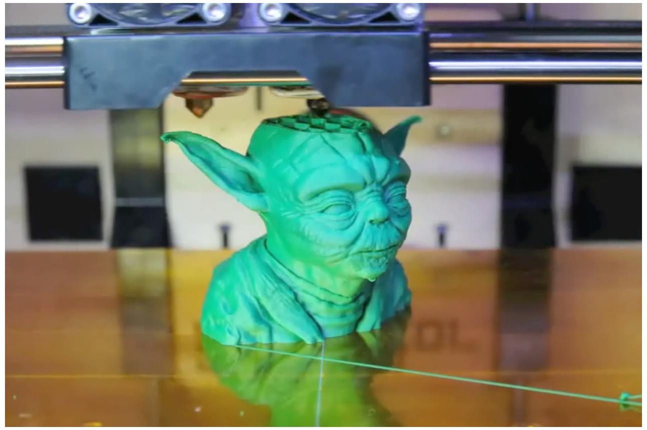
- Parts fabricability depends on the selected **digital fabrication** technique

Laser Cutting

CNC Milling



3D Printing



Low fidelity → high fidelity

Laser Cutting

- Cut flat sheets with a laser

Laser Cutting

- Fabricability: planar shapes only

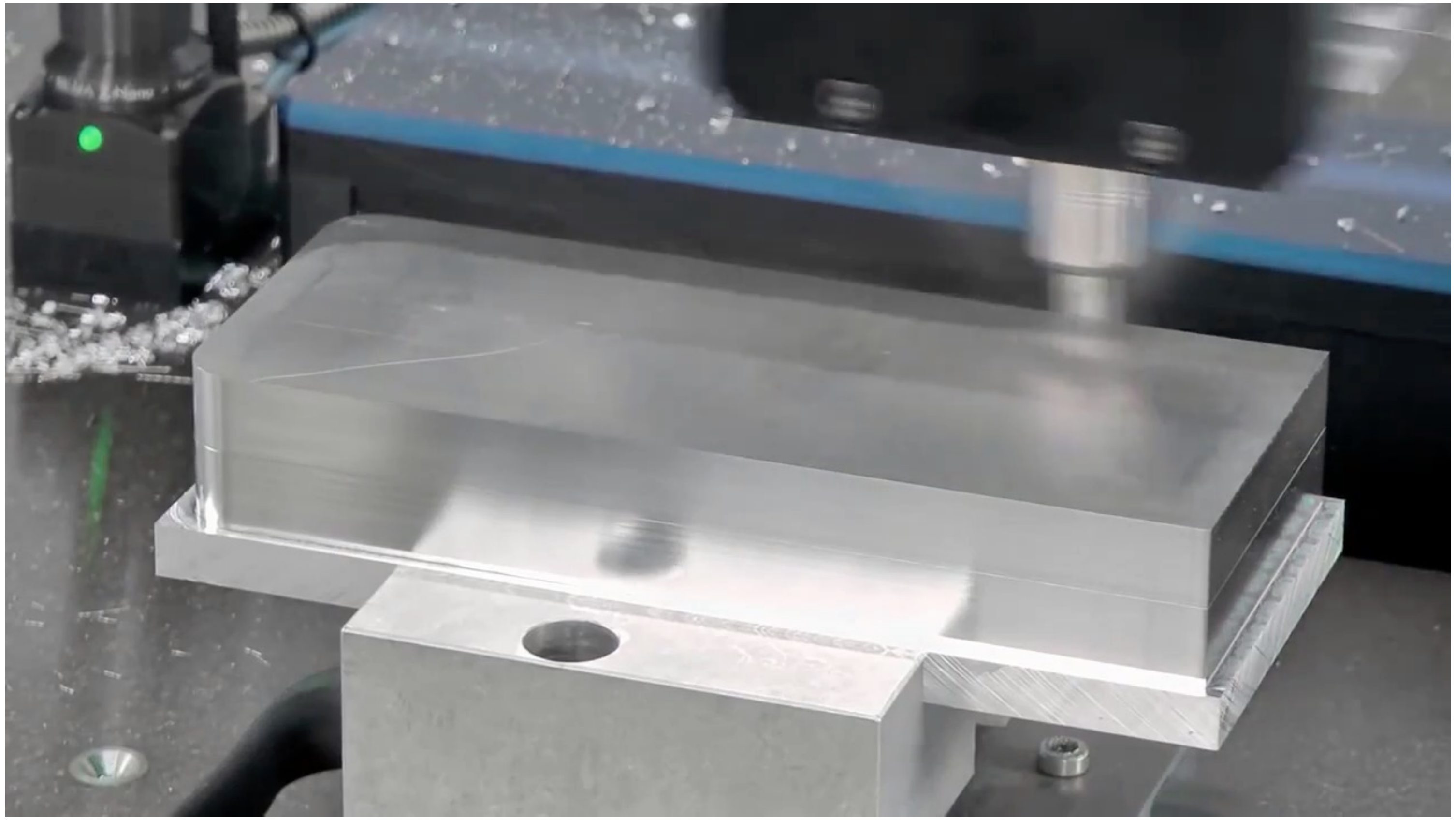
Input: 2D svg file

Output: laser cut shape

Source: [Thingiverse]

3-Axis CNC Milling

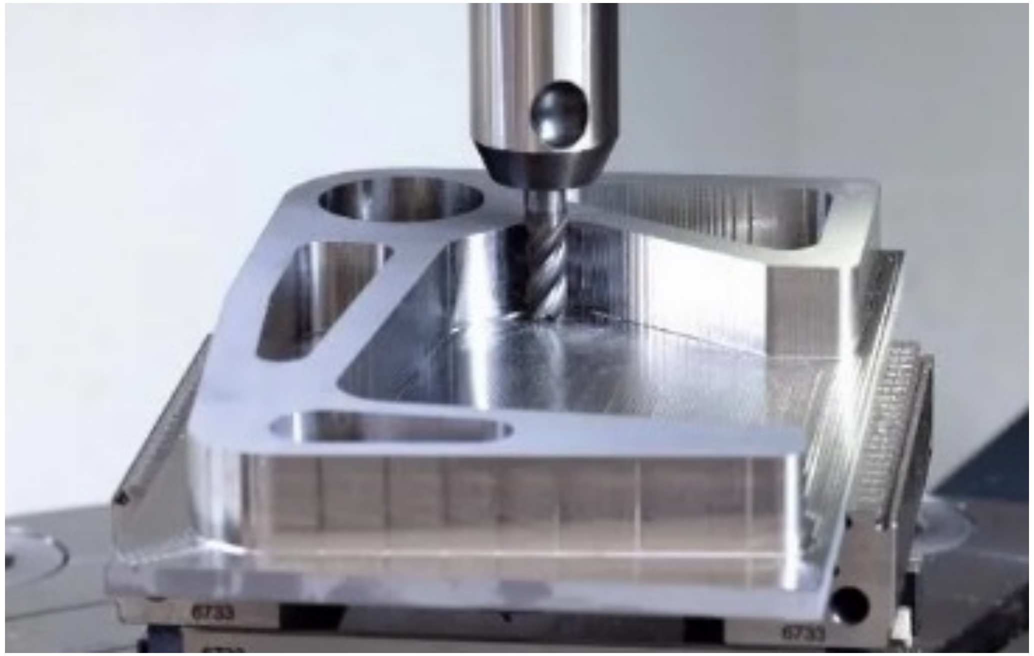
- Remove material from a starting block using a 3-axis CNC milling cutter

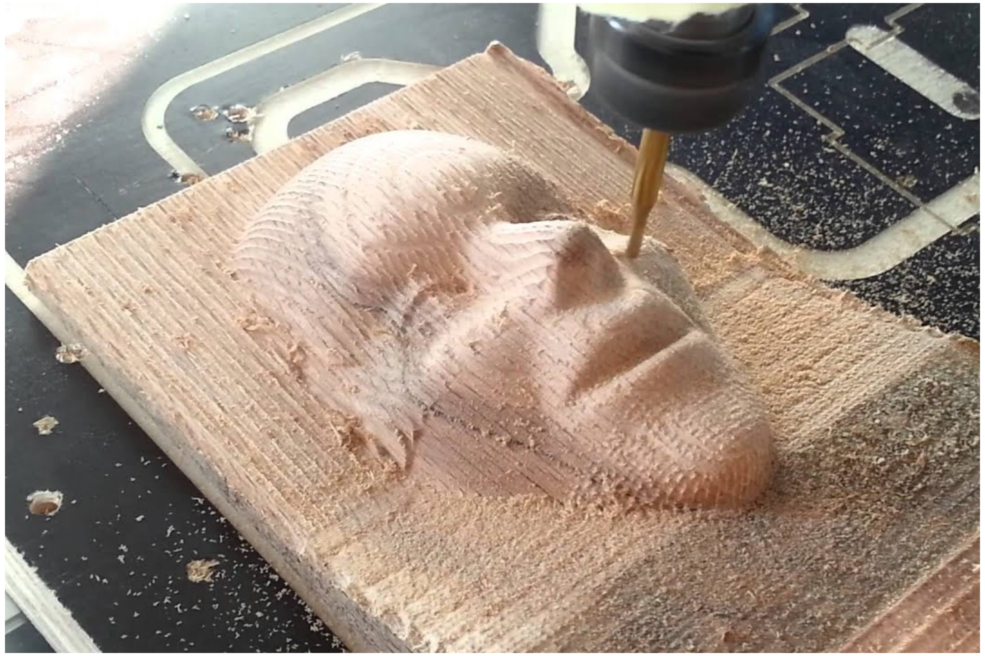


Source: youtube

3-Axis CNC Milling

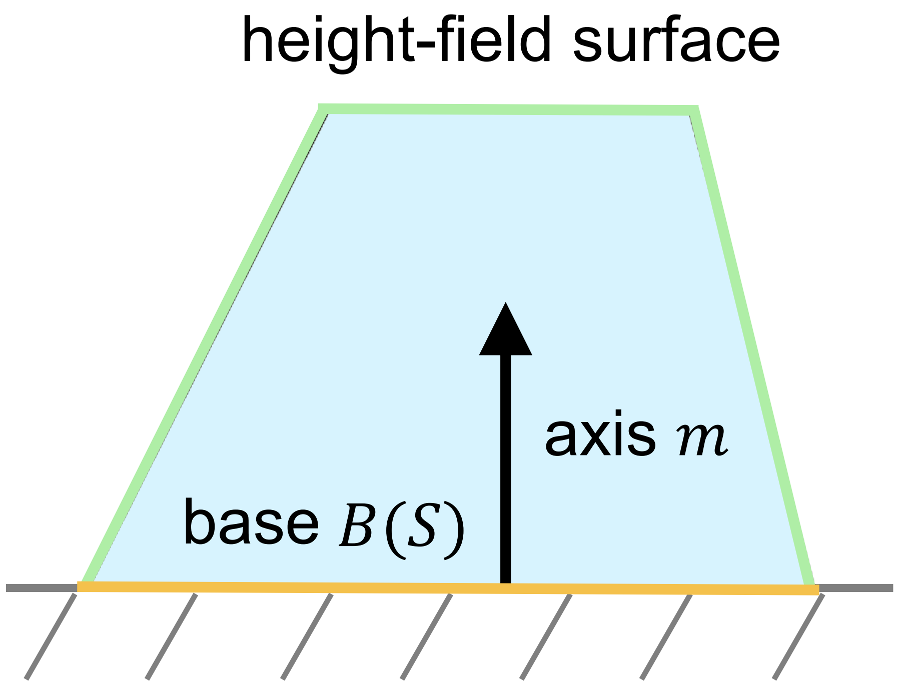
- Fabricability: 3D height-field shapes (also called pyramidal shape)

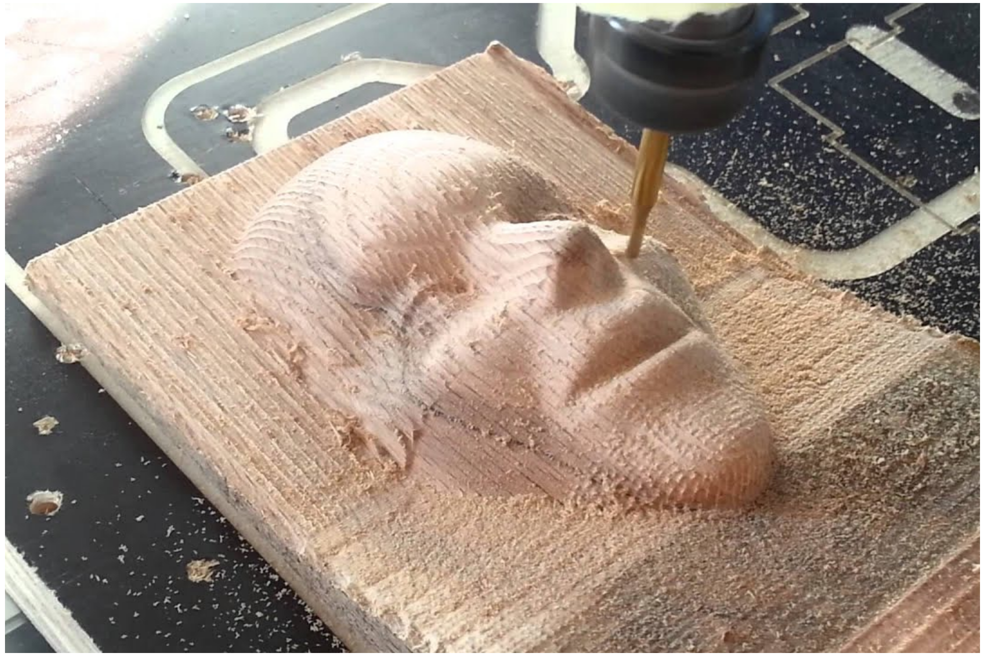




3-Axis CNC Milling

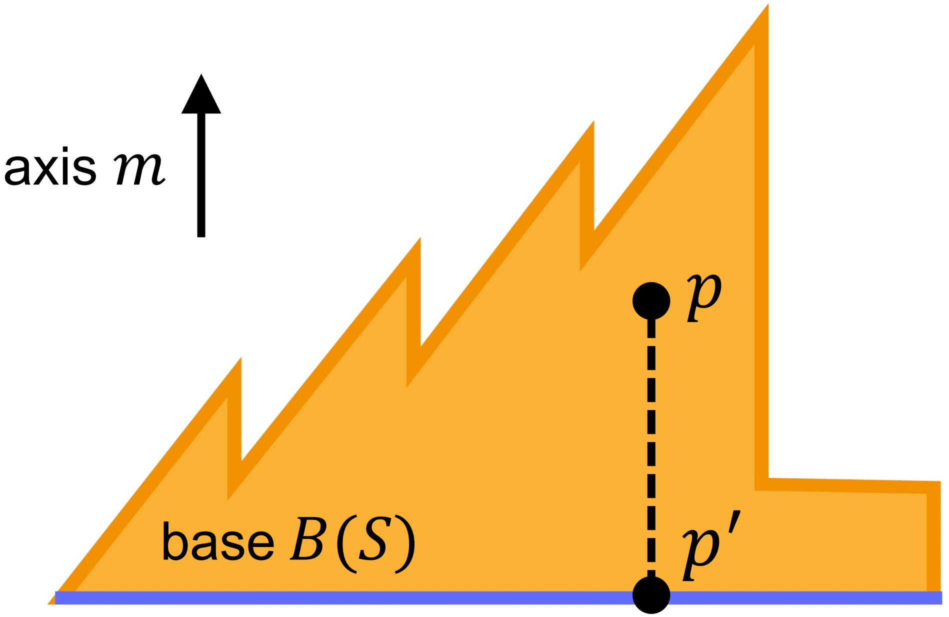
- Fabricability: 3D height-field shapes (also called pyramidal shape)





3-Axis CNC Milling

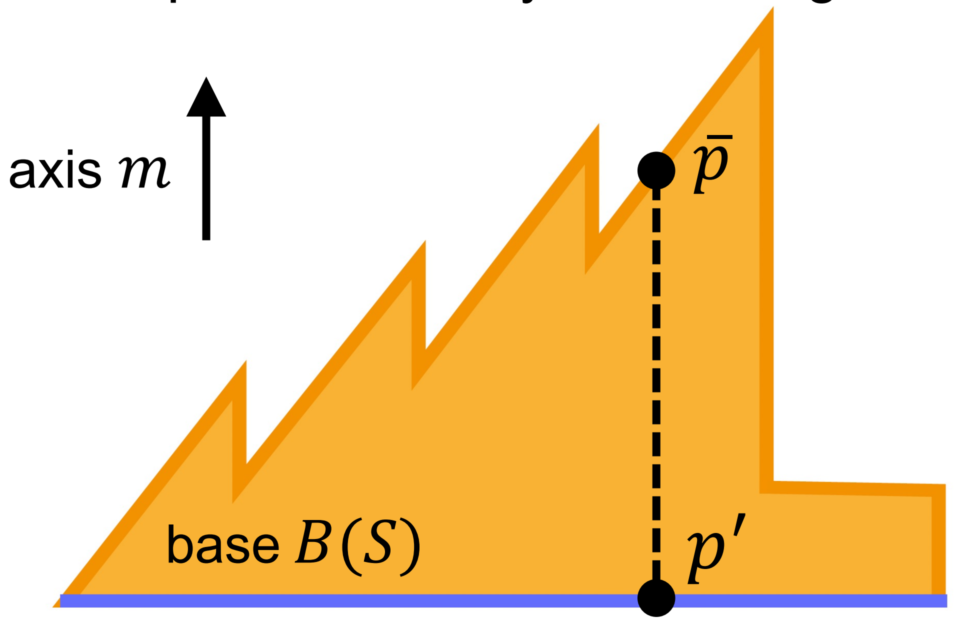
- How to test if a given shape S is a height-field shape?
- Method #1: for any point p inside S, the line segment between p and the perpendicular projection p' of p onto flat base $B(S)$ lies entirely inside S



[Hu et al. 2014]

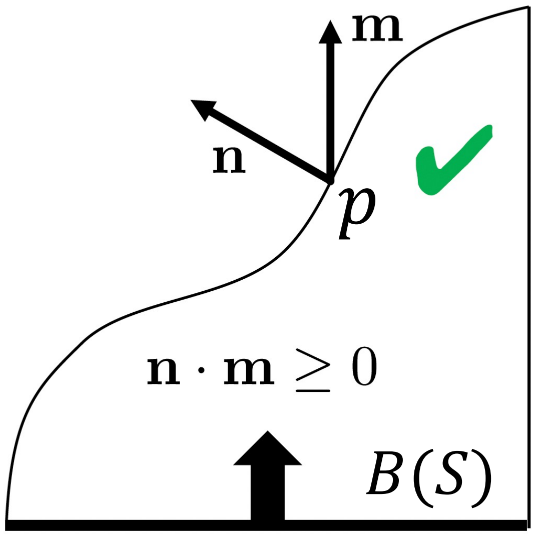
3-Axis CNC Milling

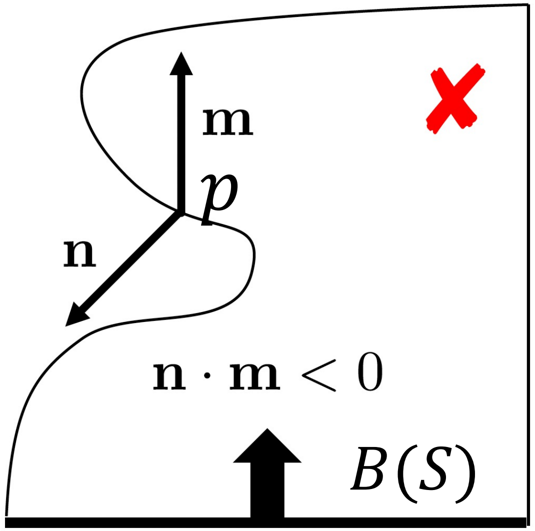
- How to test if a given shape S is a height-field shape?
- Method #2:
  1) for any point p' on the base $B(S)$, shoot a ray along axis m
  2) compute the number of intersection points between the ray and the shape boundary
  3) the number of intersection points is always 1 for height-field shapes



3-Axis CNC Milling

- How to test if a given shape S is a height-field shape?
- Method #3: for any point p on the non-base surface, the angle between the outward pointing normal n of point p and the base direction m to be acute

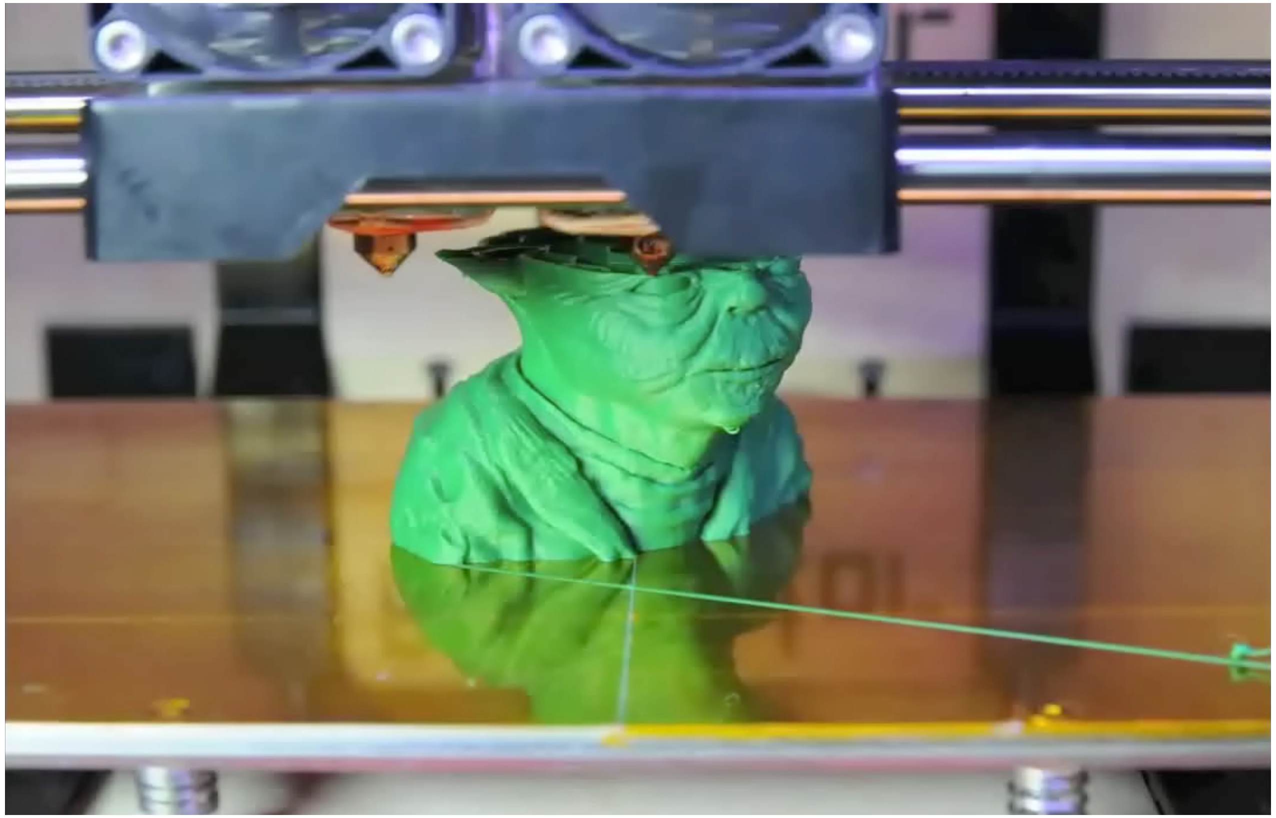




[Muntoni et al. 2018]

3D Printing

- Slice a 3D object into many layers
- Fabricate layers one at a time, from bottom to top



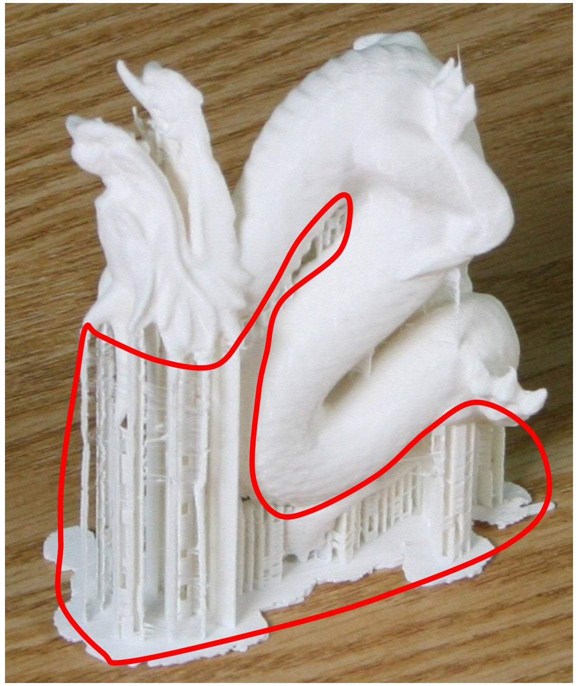
3D Printing

- Fabricability: 3D shapes of almost any complexity

source: Ozeki

3D Printing

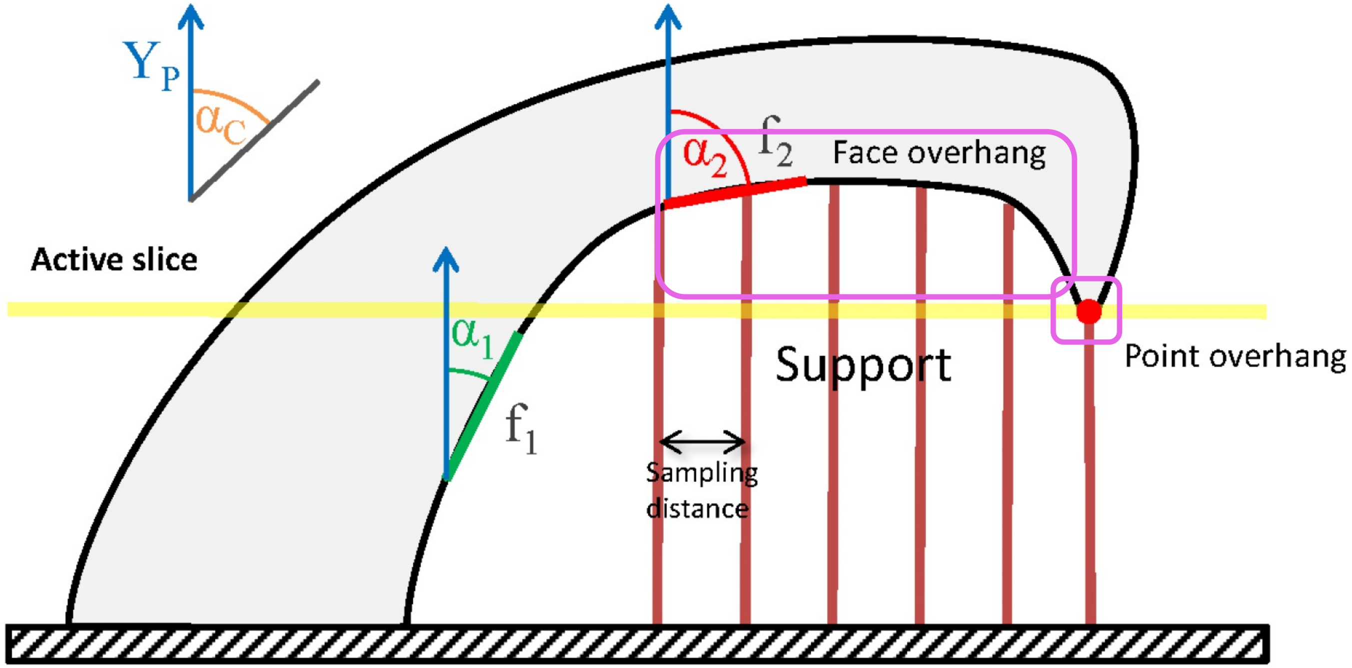
- Support materials are required for 3D printing shapes with **overhangs**.
  - require more materials and more printing time
  - peeling away support materials may affect the appearance



[Vanek et al. 2014]

3D Printing

- Overhang detection
  - Point overhang: point lower than its neighbors
  - Face overhang: face normals that the deviation angle $\alpha > \alpha_c$



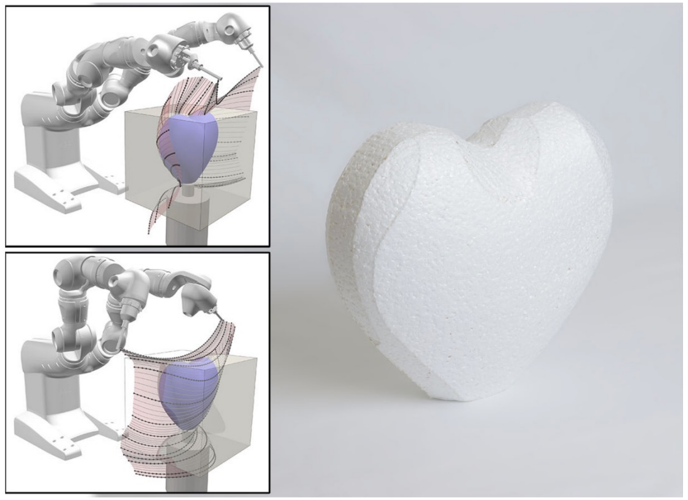
[Vanek et al. 2014]

Research on Parts Fabricability

- Study new digital fabrication techniques
  - enable making novel shapes
  - enable better fabrication performance

Hot-wire Cutting

Sweep surface



[Duenser et al. 2020]

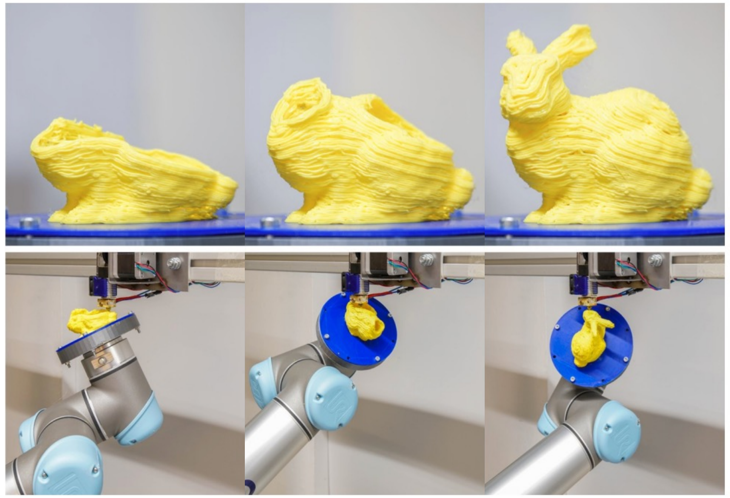
2-pass 3-axis CNC Milling

Double height-field shape

[Yang et al. 2020]

6-DOF 3D Printing

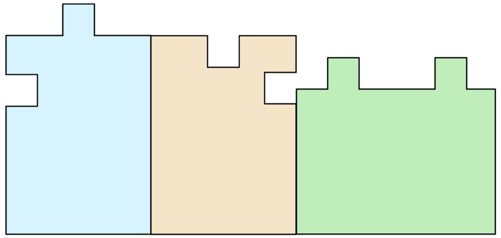
Support-free shape



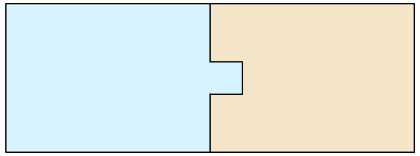
[Dai et al. 2018]

Computational Analysis of Assemblies

Parts fabricability

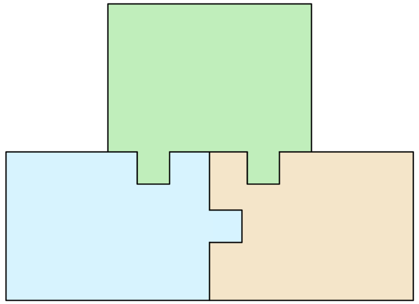


Parts joining



Assembly planning

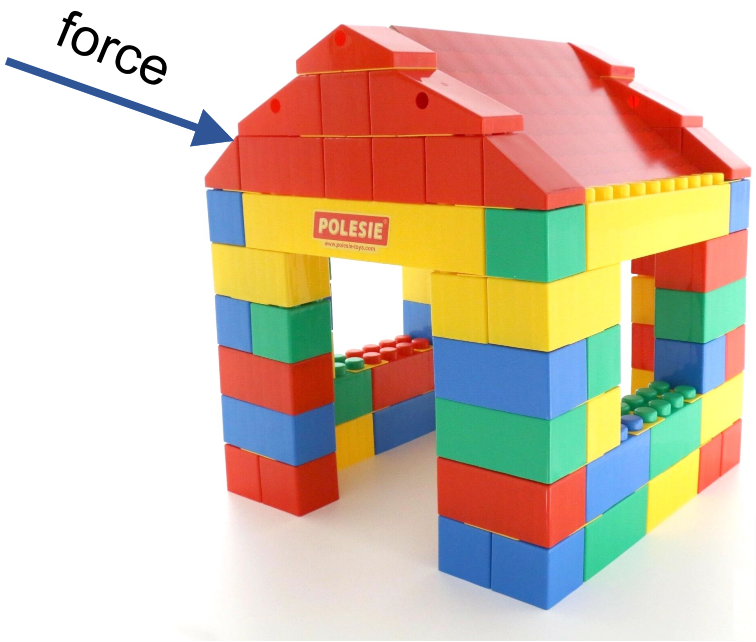
Structural stability



Peng Song Ziqi Wang Marco Livesu

Parts Joining

- In an assembly, component parts need to be joined together to make the assembly stable.



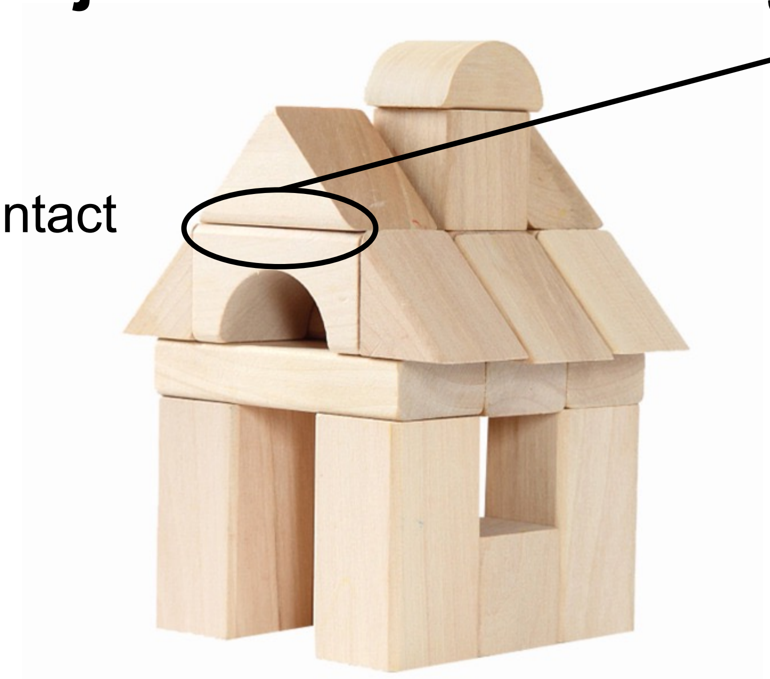
More stable!

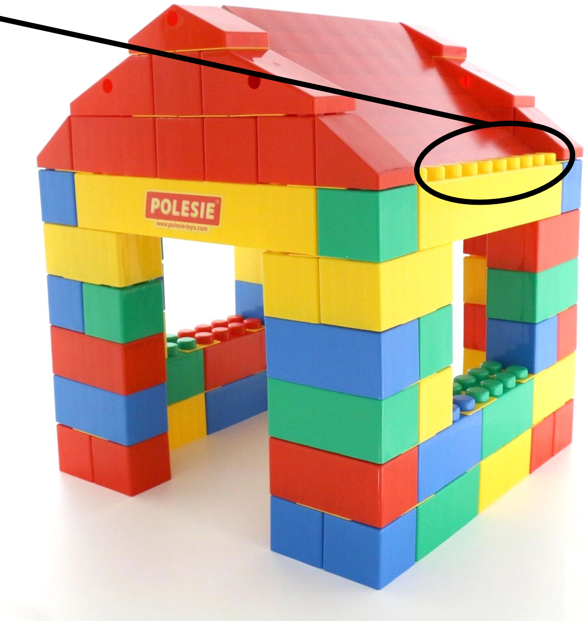
Parts Joining

- In an assembly, component parts need to be joined together to make the assembly stable.
- Geometry or material used to connect parts defines the joining method, or simply the **joint**

Joint

Planar contact





Friction joint

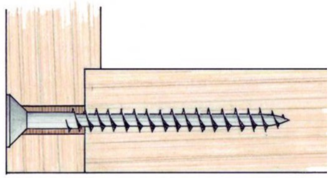
Joint Classification

Joints

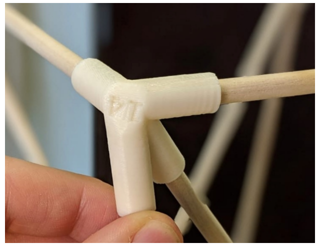
- Non-permanent
- Permanent

Non-permanent:

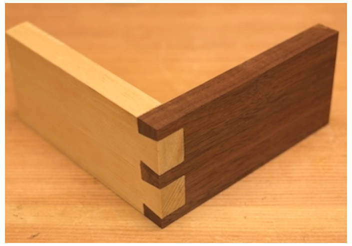
screw



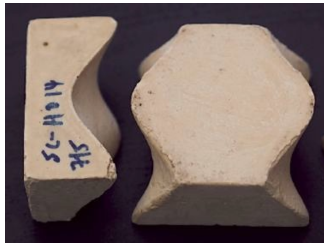
3D printable



woodworking



curved-contact



Permanent:

glue

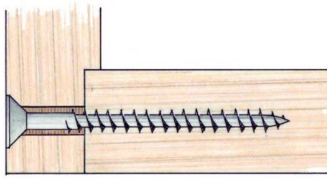
riveting

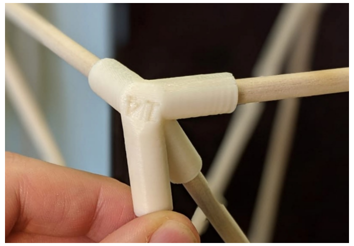
Peng Song Ziqi Wang Marco Livesu

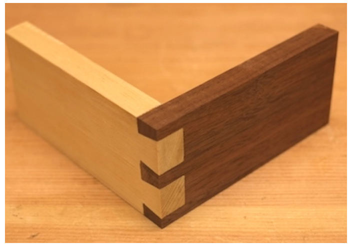
Joint Classification

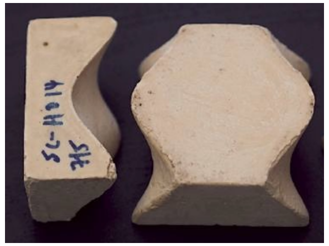
Joints

- Non-permanent
  - External
  - Integral
- Permanent







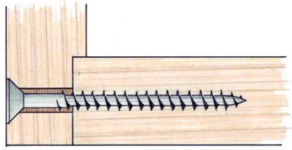


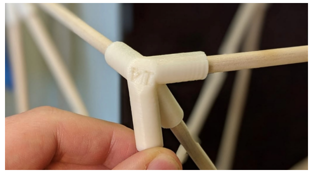
Advantages of integral joints:

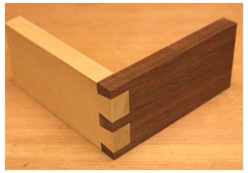
1. Reduce the total number of assembly steps
2. Assist parts alignment for assembly

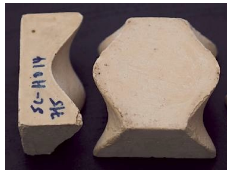
Joint Classification

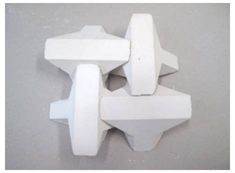
- Joints
  - Non-permanent
    - External
      - Conventional
      - Customized
    - Integral
      - Woodworking
      - Curved contact
      - Planar contact
  - Permanent







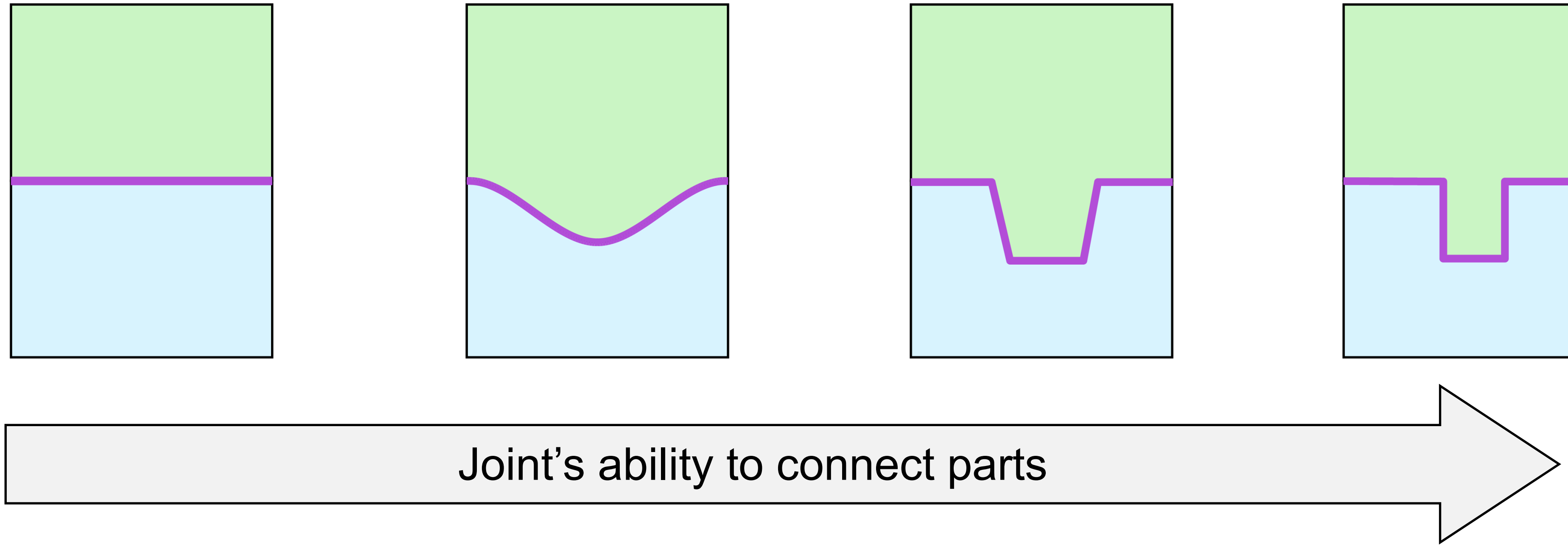




Peng Song Ziqi Wang Marco Livesu

Joint Mobility Analysis

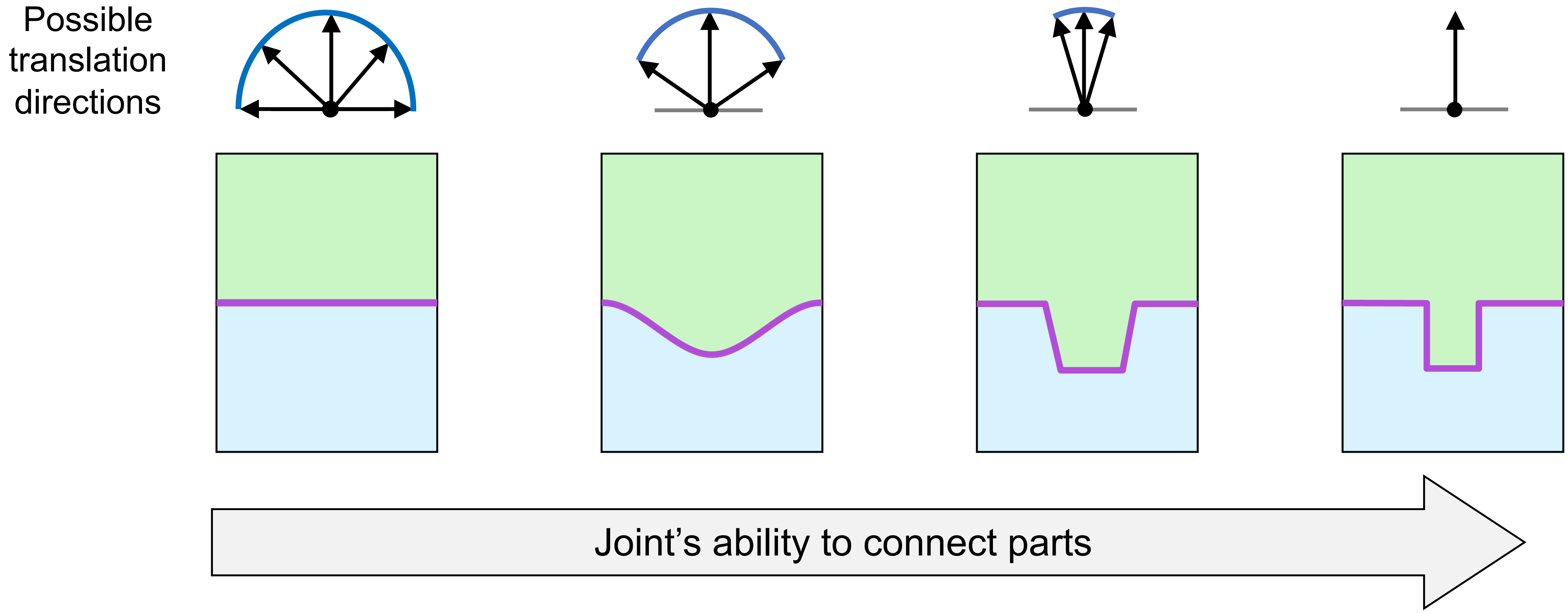
- Each joint connects two parts by restricting their relative movement



[Wang et al. 2021]

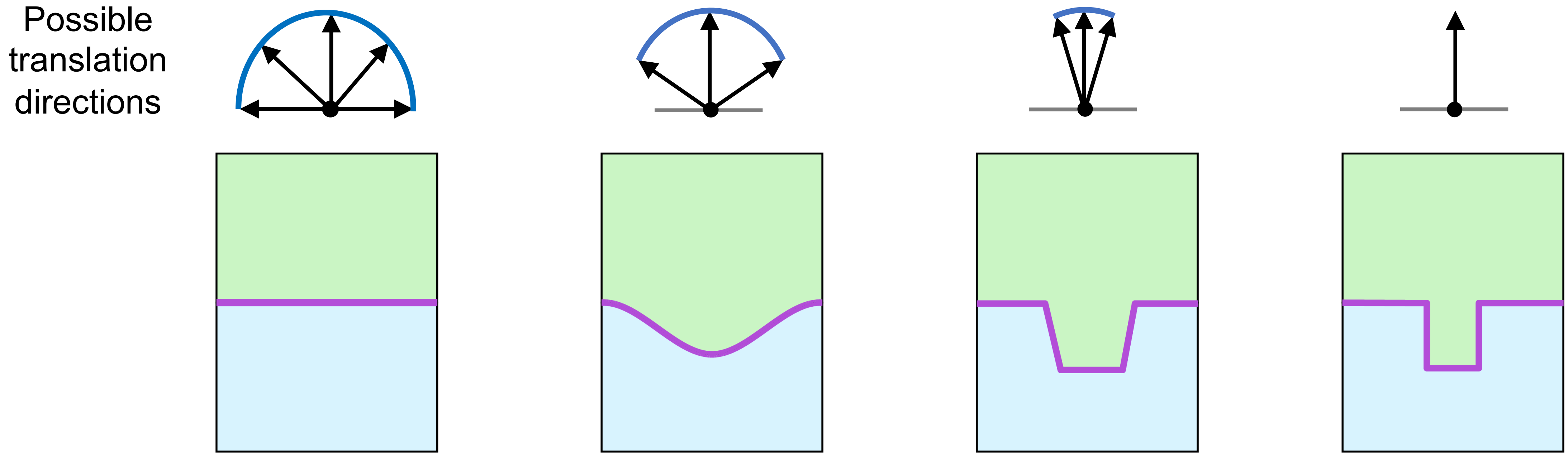
Joint Mobility Analysis

- Each joint connects two parts by restricting their relative movement
- Ability to connect parts = Ability to restrict part relative movement



Joint Mobility Analysis

- Each joint connects two parts by restricting their relative movement
- Ability to connect parts = Ability to restrict part relative movement

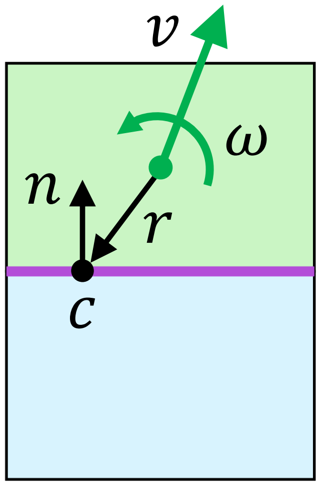


Question: how to compute the above part motion space, especially for rigid motion?

Joint Mobility Analysis

- Given joint geometry, quantitatively compute the part motion space
  - Assume one part is fixed, calculate the motion space of the other one
  - Constraint: avoid collision during the part movement

- Case 1: planar contact



The constraint for a contact point c

$$v_c \cdot n \geq 0$$

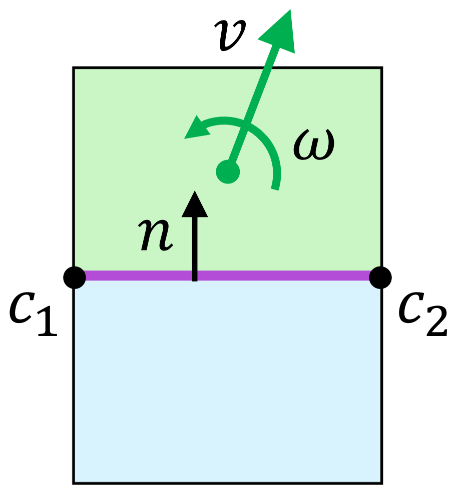
where

$$v_c = v + \omega \times r$$

Joint Mobility Analysis

- Given joint geometry, quantitatively compute the part motion space
  - Assume one part is fixed, calculate the motion space of the other one
  - Constraint: avoid collision during the part movement

- Case 1: planar contact



The constraint for a planar contact

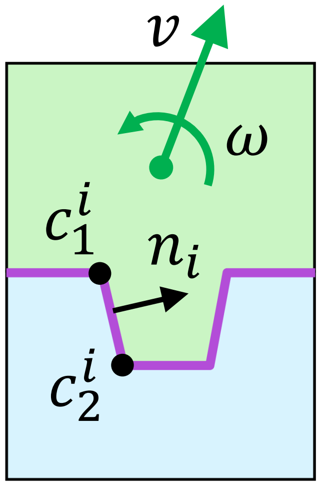
$$v_{c1} \cdot n \geq 0$$

$$v_{c2} \cdot n \geq 0$$

Joint Mobility Analysis

- Given joint geometry, quantitatively compute the part motion space
  - Assume one part is fixed, calculate the motion space of the other one
  - Constraint: avoid collision during the part movement

- Case 2: multiple planar contacts



The constraint for a planar-contact joint

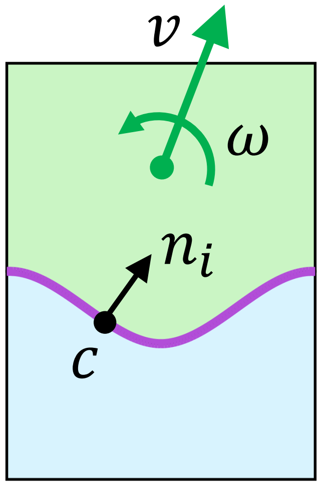
$$\begin{bmatrix} \vdots \\ v_{c1}^i \cdot n_i \\ v_{c2}^i \cdot n_i \\ \vdots \end{bmatrix} \geq 0$$

joint motion space = solution of the linear system

Joint Mobility Analysis

- Given joint geometry, quantitatively compute the part motion space
  - Assume one part is fixed, calculate the motion space of the other one
  - Constraint: avoid collision during the part movement

- Case 3: curved contact



The constraint for a curved-contact joint

$$\begin{bmatrix} \vdots \\ v_c^i \cdot n_i \\ \vdots \end{bmatrix} \geq 0$$

The computed motion space is an **upper bound** of the actual motion space

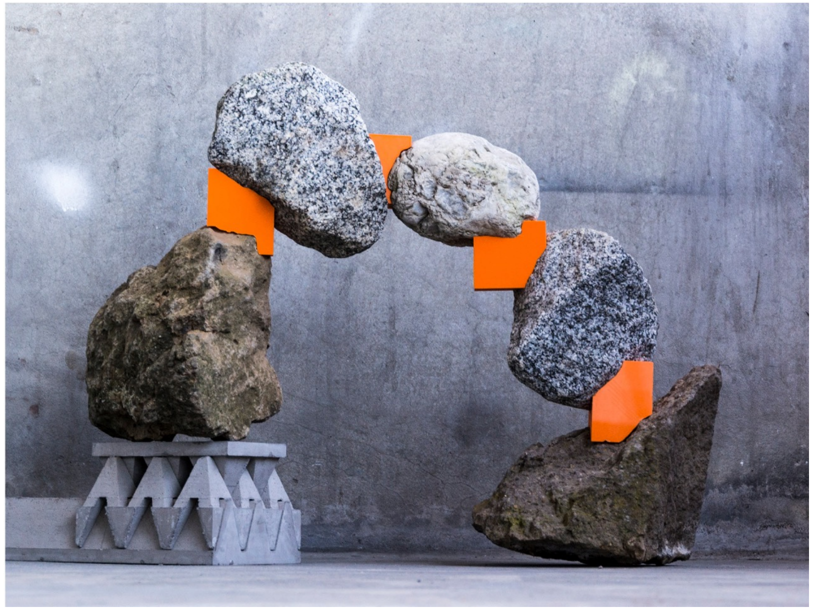
Research on Parts Joining

- Study customized **external** joints for connecting a set of given parts/objects

2-way joint

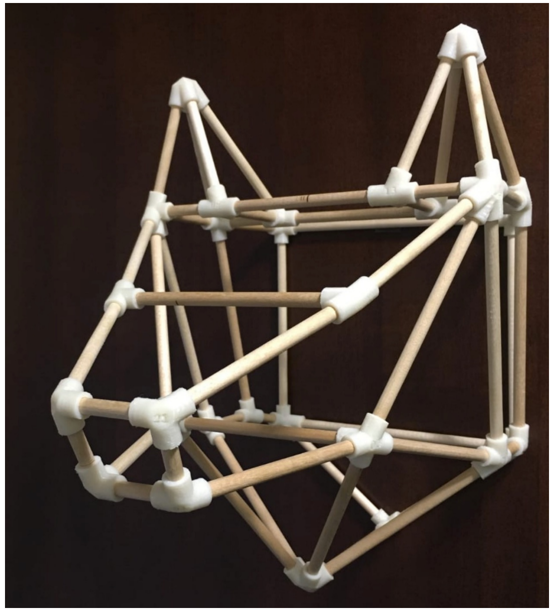
[Koyama et al. 2015]

2-way joint



[Wibranek et al. 2019]

N-way joint



[Jacobson 2019]

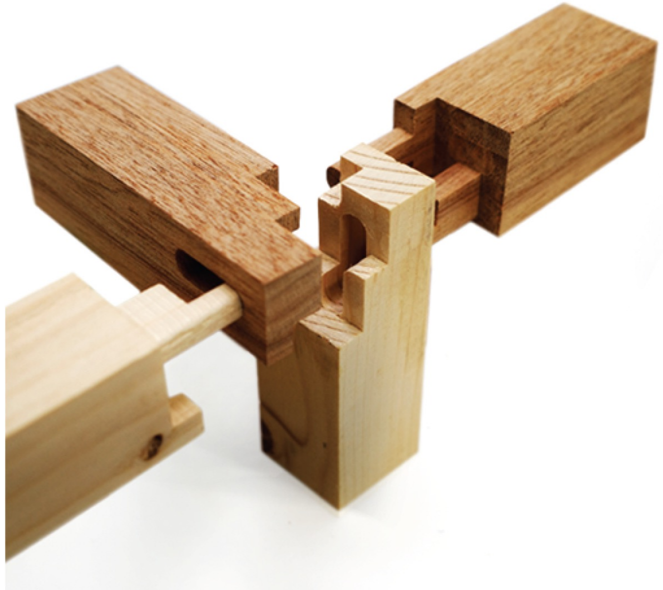
Research on Parts Joining

- Study customized **integral** joints for different purposes

decoration

[Yao et al. 2017]

structural stability



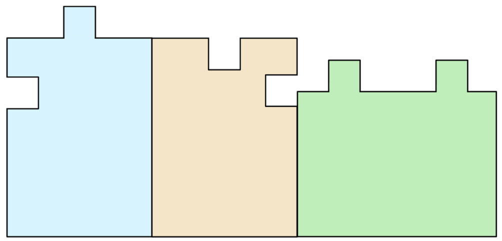
[Tsugite et al. 2020]

structural stability & ease-of-assembly

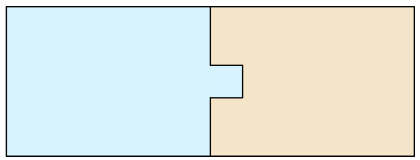
[Wang et al. 2021]

Computational Analysis of Assemblies

Parts fabricability

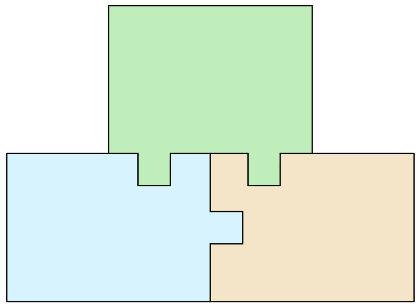


Parts joining



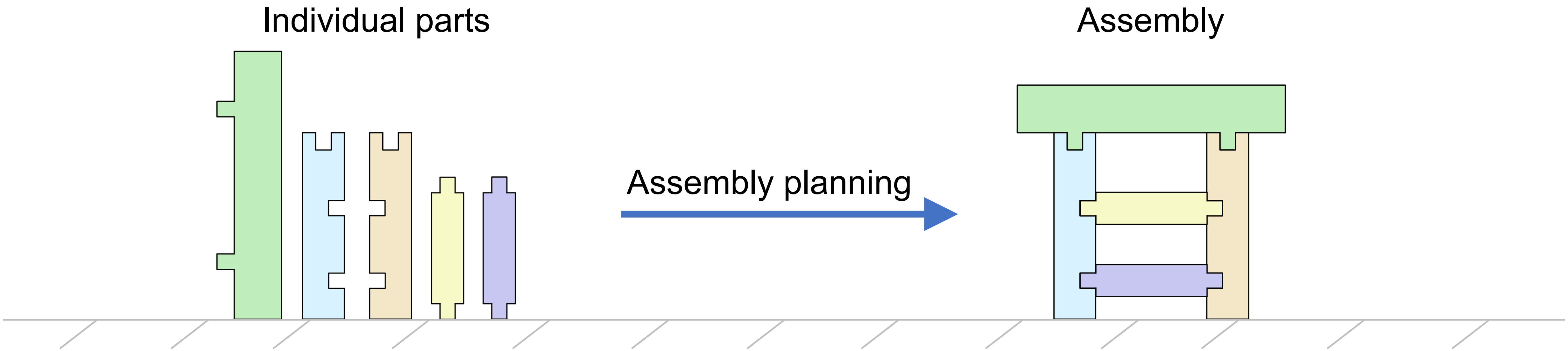
Assembly planning

Structural stability



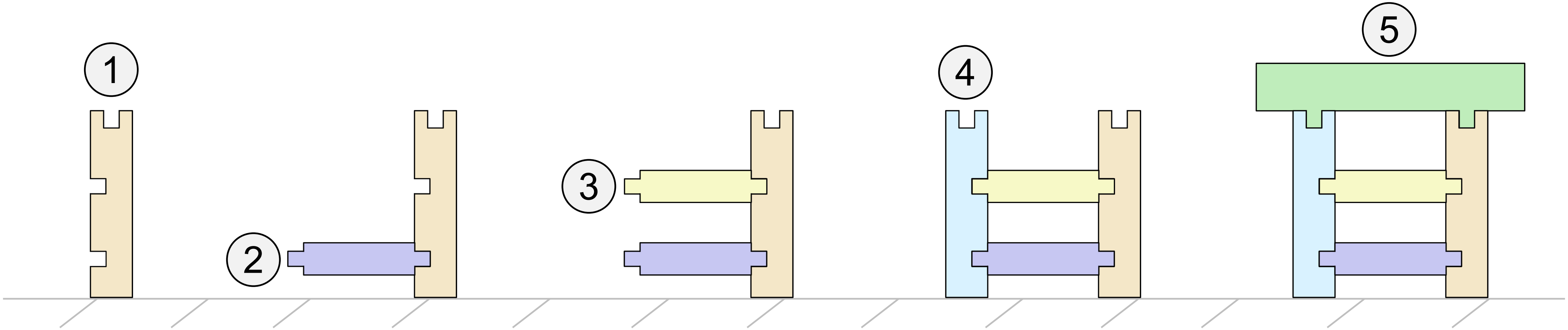
Assembly Planning

- Assembly planning: create instructions to combine separate parts into the final assembly



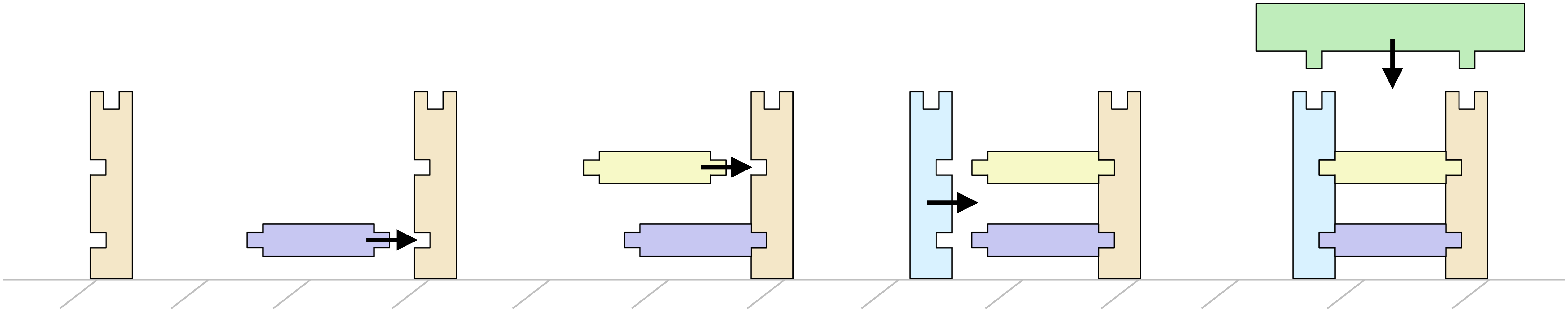
Assembly Planning

- Assembly planning: create instructions to combine separate parts into the final assembly
  1. Order to assemble the parts (assembly sequencing)



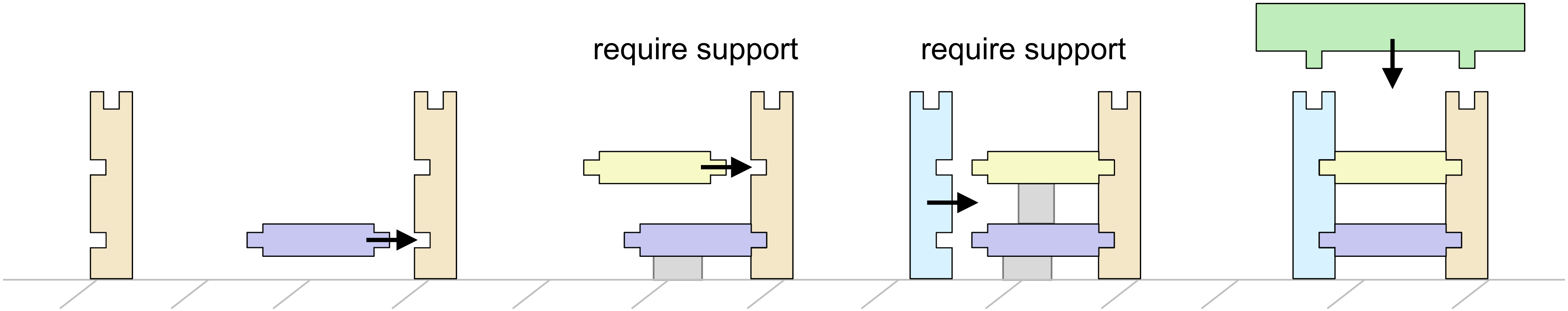
Assembly Planning

- Assembly planning: create instructions to combine separate parts into the final assembly
  1. Order to assemble the parts (assembly sequencing)
  2. Motion to bring each part to its target pose (assembly path planning)



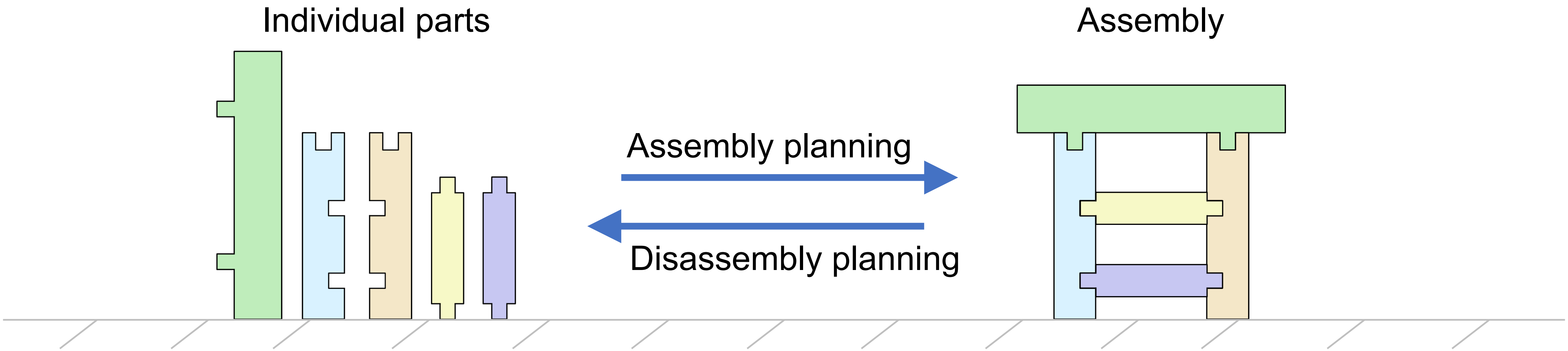
Assembly Planning

- Assembly planning: create instructions to combine separate parts into the final assembly
  1. Order to assemble the parts (assembly sequencing)
  2. Motion to bring each part to its target pose (assembly path planning)
  3. Utilization of additional resources such as supports and tools



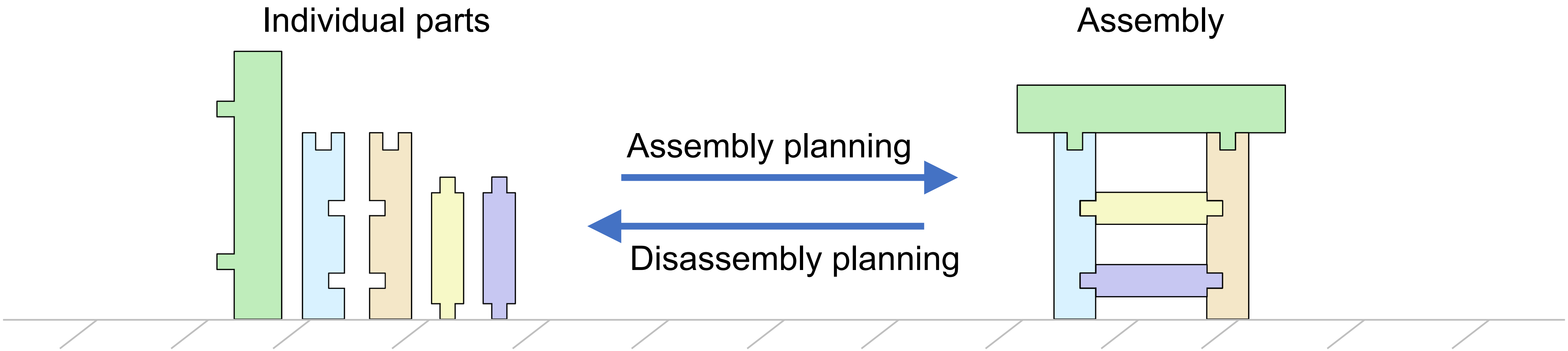
Disassembly Planning

- Disassembly planning: create instructions to disassemble parts from an installed assembly.
- A bijection exists between assembly and disassembly sequences and paths
  - Assumption: only when geometric constraints are concerned



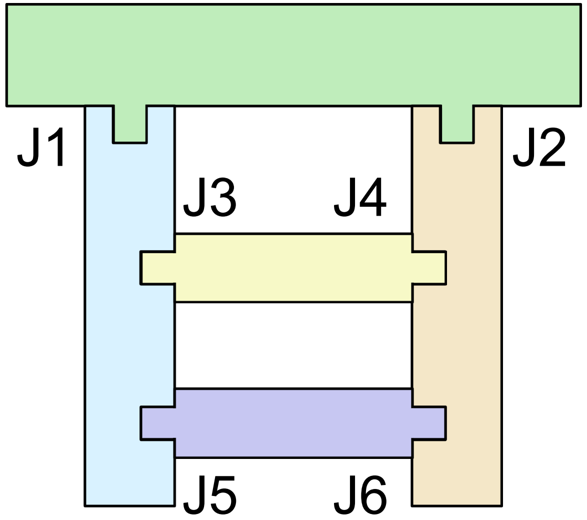
Assembly by Disassembly

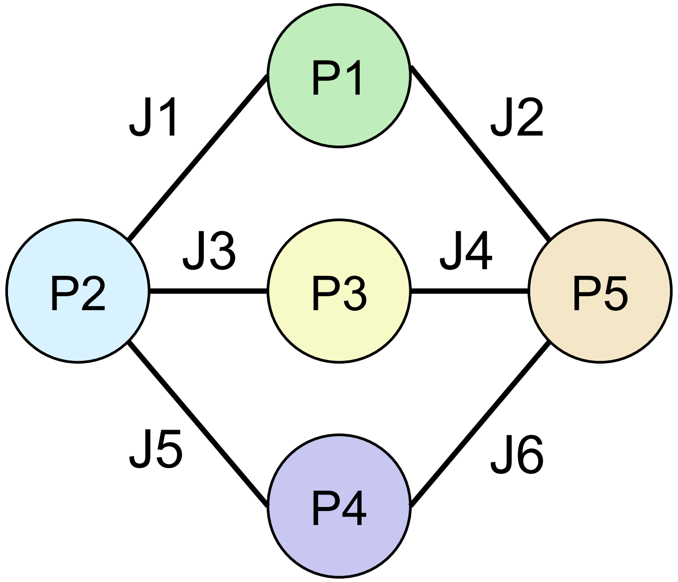
- Assembly by disassembly strategy
  - Given an assembly, first compute a disassembly plan
  - Reverse the disassembly plan to obtain an assembly plan
- Advantage: drastically reduce the size of the solution space



Search for a Valid Disassembly Plan

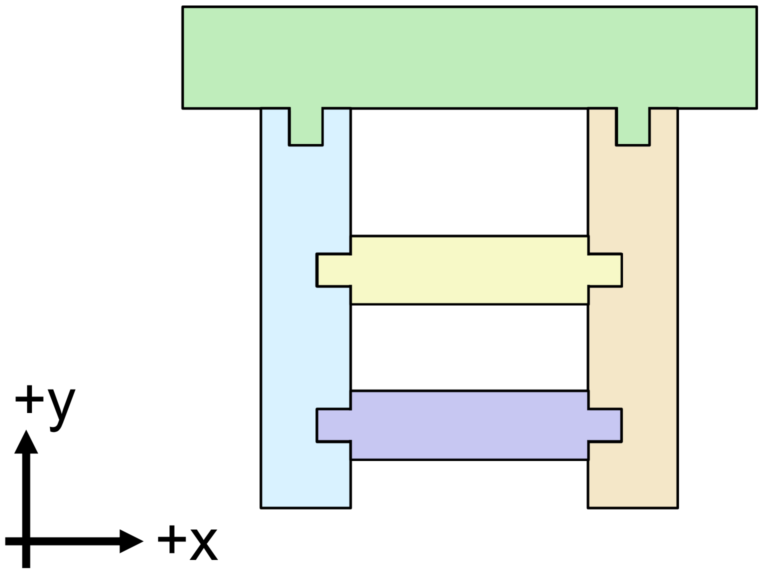
1. Build a *parts-graph* for a given assembly
   - each node represents a part
   - each edge represents joints between the two parts

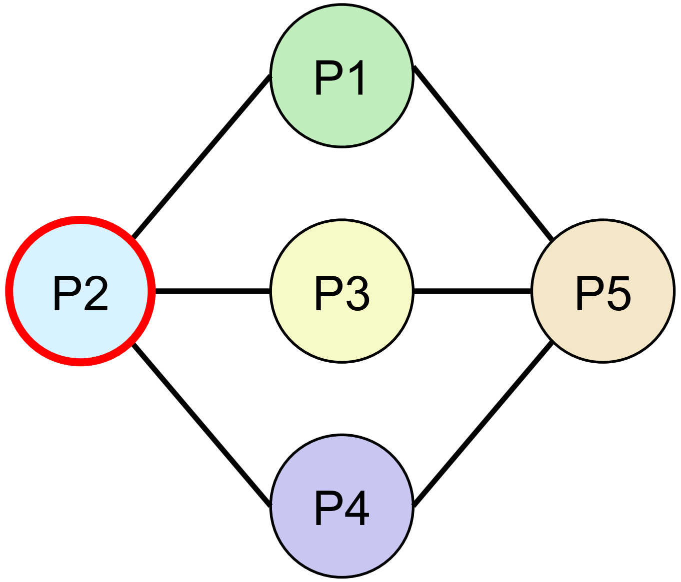




Search for a Valid Disassembly Plan

1. Build a *parts-graph* for a given assembly
2. Compute mobility $M(P_i)$ for each part P_i
   - Identify all the edges (joints) $\{J_k\}$ associated with P_i
   - $M(P_i) = \bigcap M(J_k)$

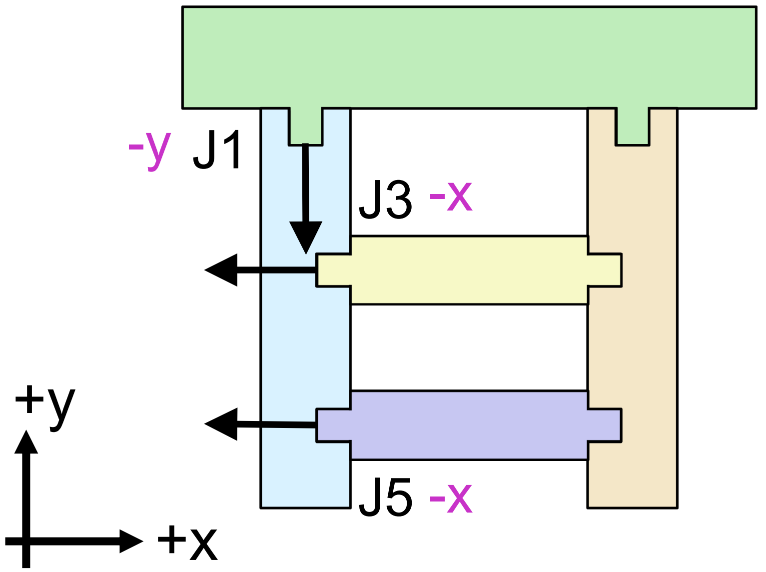


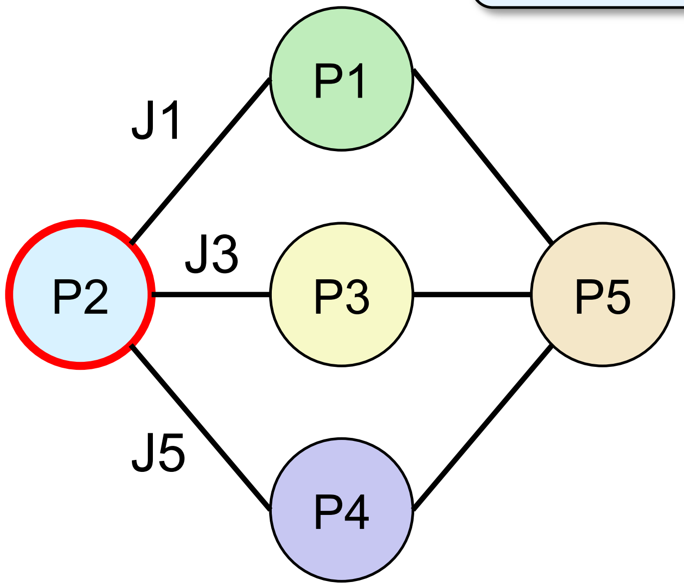


Search for a Valid Disassembly Plan

1. Build a *parts-graph* for a given assembly
2. Compute mobility $M(P_i)$ for each part P_i
   - Identify all the edges (joints) $\{J_k\}$ associated with P_i
   - $M(P_i) = \cap\, M(J_k)$

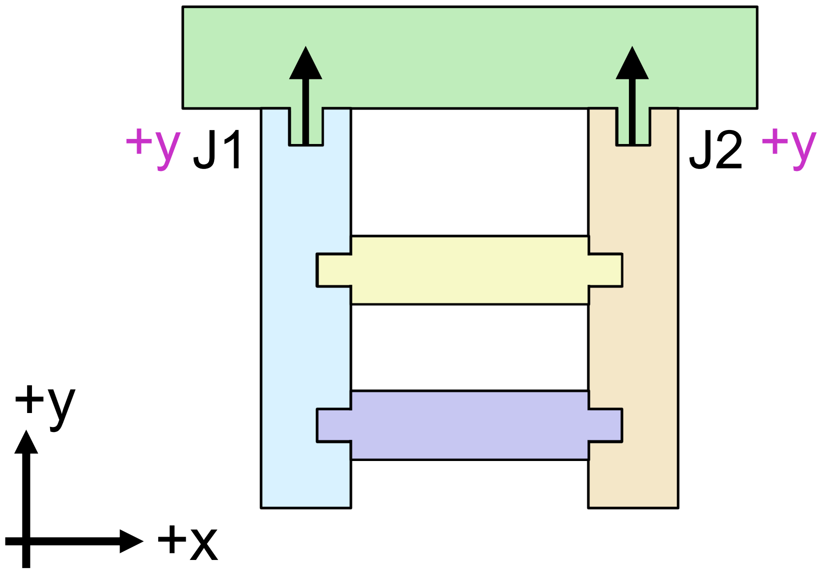
$$M(P_2) = \{-y\} \cap \{-x\} \cap \{-x\} = \emptyset$$

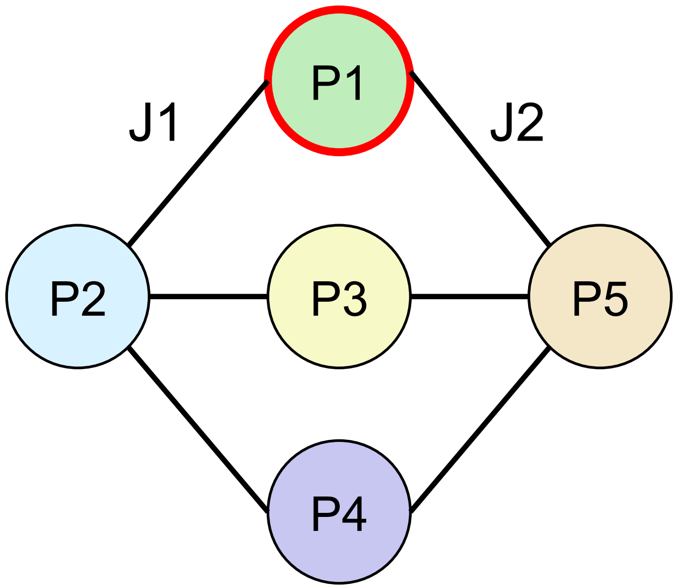




Search for a Valid Disassembly Plan

1. Build a *parts-graph* for a given assembly
2. Compute mobility $M(P_i)$ for each part P_i
   - Identify all the edges (joints) $\{J_k\}$ associated with P_i
   - $M(P_i) = \cap M(J_k)$

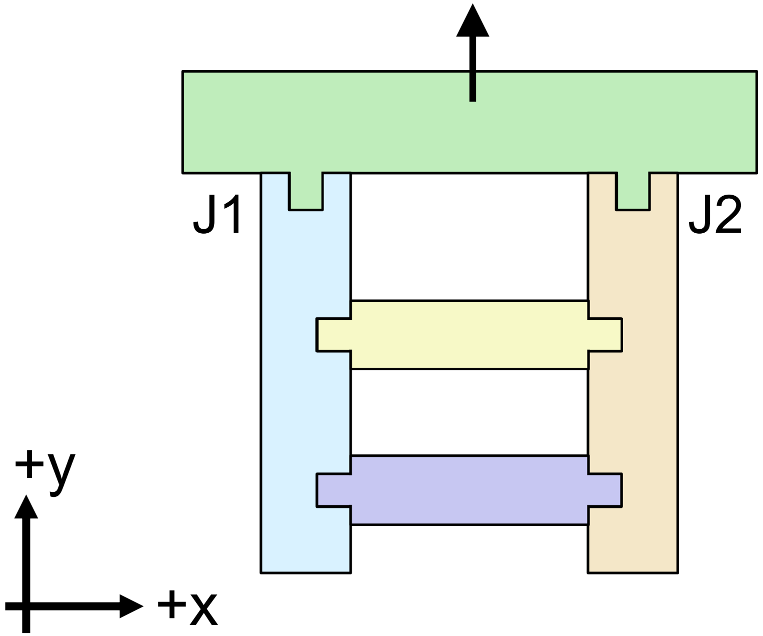


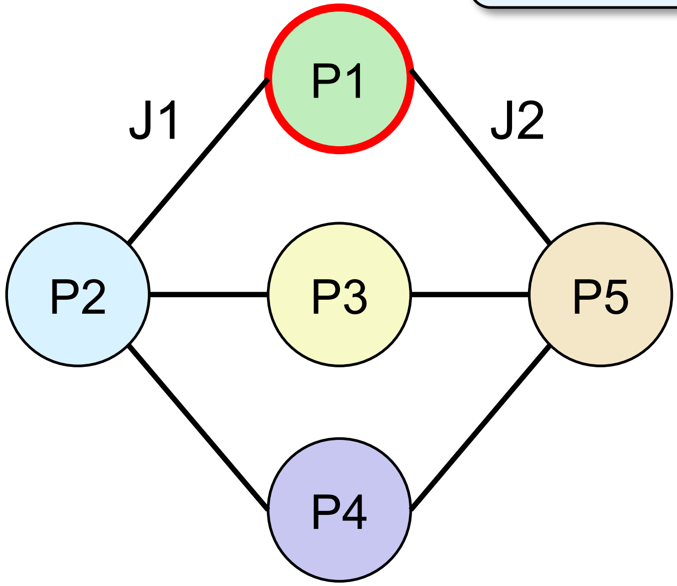


Search for a Valid Disassembly Plan

1. Build a *parts-graph* for a given assembly
2. Compute mobility $M(P_i)$ for each part P_i
   - Identify all the edges (joints) $\{J_k\}$ associated with P_i
   - $M(P_i) = \bigcap M(J_k)$

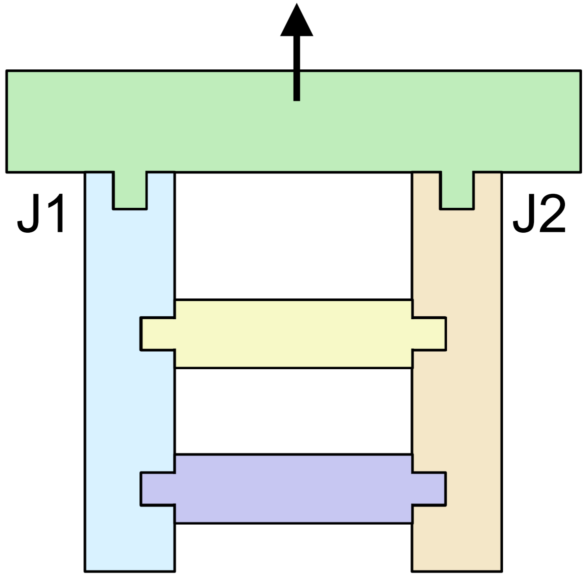
$$M(P_1) = \{+y\} \cap \{+y\} = \{+y\}$$

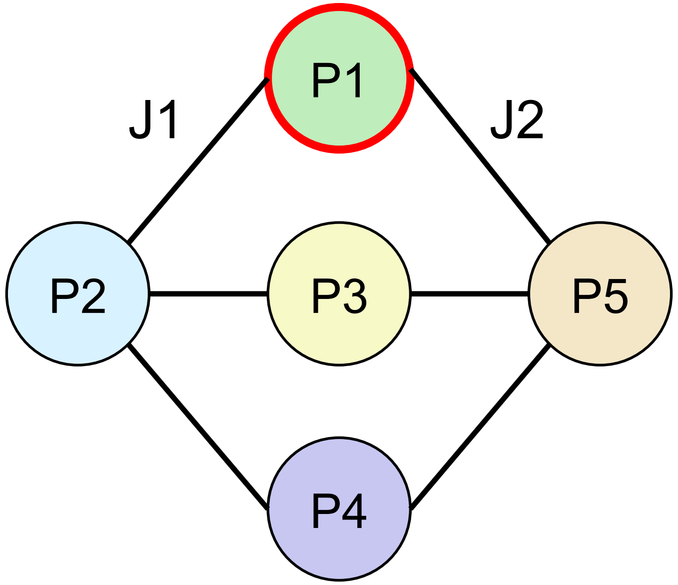




Search for a Valid Disassembly Plan

1. Build a *parts-graph* for a given assembly
2. Compute mobility $M(P_i)$ for each part P_i
3. Select a moveable part

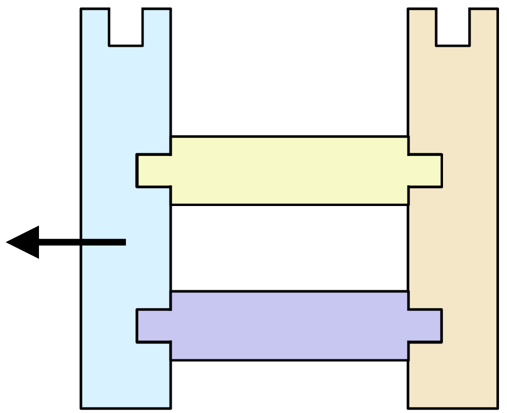


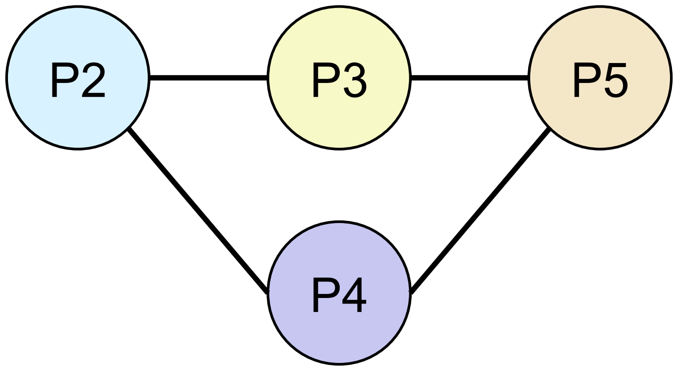


Search for a Valid Disassembly Plan

1. Build a *parts-graph* for a given assembly
2. Compute mobility $M(P_i)$ for each part P_i
3. Select a moveable part
4. Remove the part and update the graph

Repeat steps 3-4

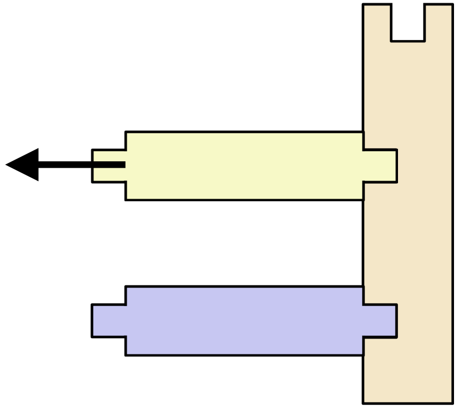


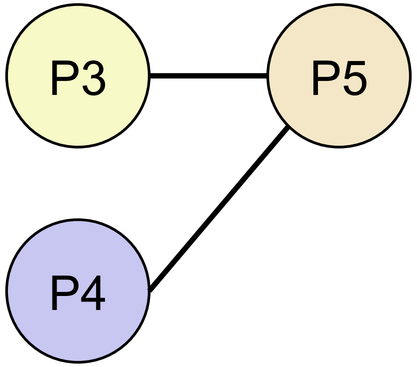


Search for a Valid Disassembly Plan

1. Build a *parts-graph* for a given assembly
2. Compute mobility $M(P_i)$ for each part P_i
3. Select a moveable part
4. Remove the part and update the graph

Repeat steps 3-4

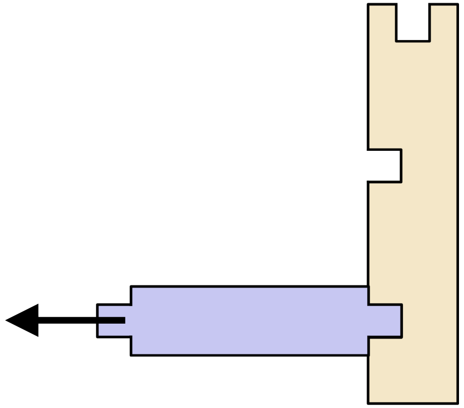


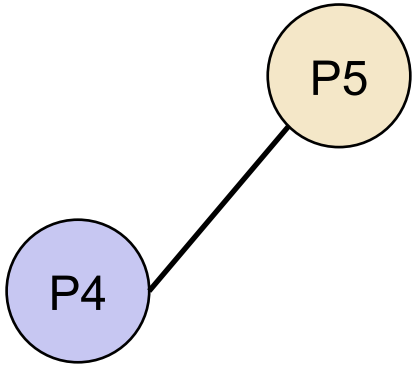


Search for a Valid Disassembly Plan

1. Build a *parts-graph* for a given assembly
2. Compute mobility $M(P_i)$ for each part P_i
3. Select a moveable part
4. Remove the part and update the graph

Repeat steps 3-4

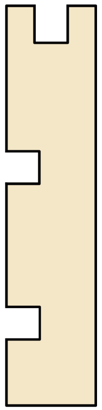


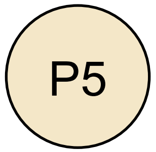


Search for a Valid Disassembly Plan

1. Build a *parts-graph* for a given assembly
2. Compute mobility $M(P_i)$ for each part P_i
3. Select a moveable part
4. Remove the part and update the graph

Until only one part left in the assembly

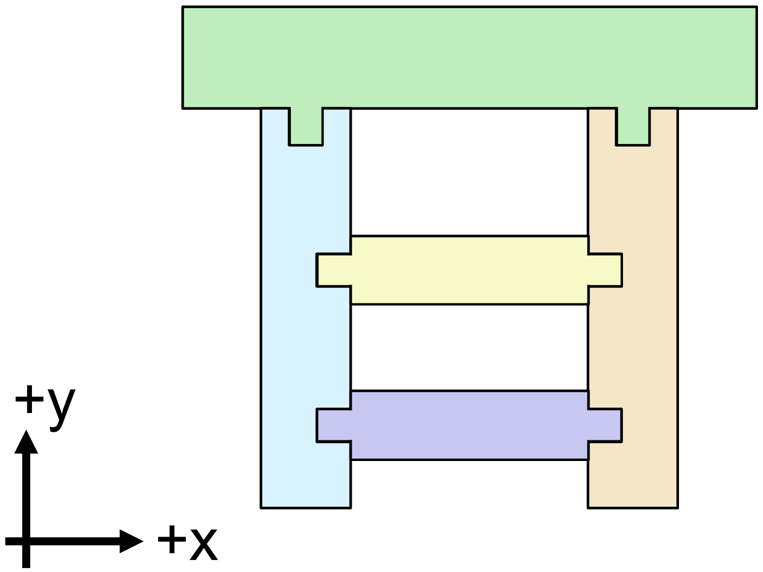


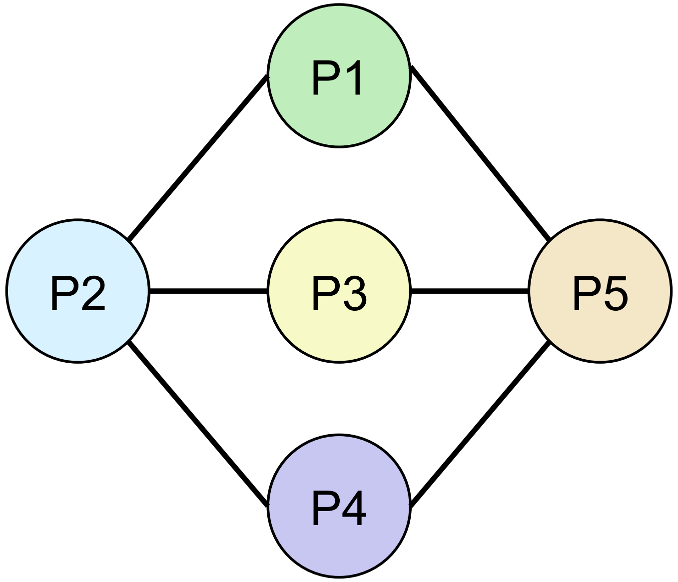


Search for a Valid Disassembly Plan

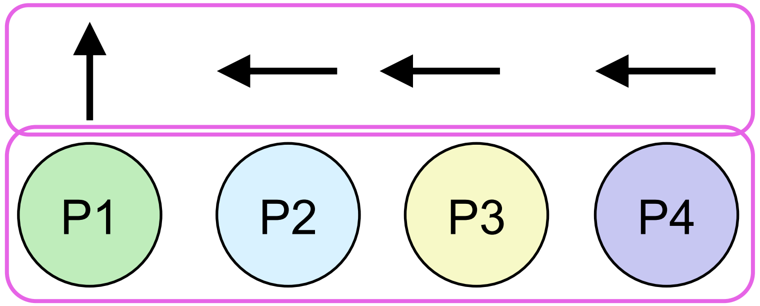
1. Build a *parts-graph* for a given assembly
2. Compute mobility $M(P_i)$ for each part P_i
3. Select a moveable part
4. Remove the part and update the graph

Assumptions:
1. Move a single part at a time
2. Move parts in a sequence
3. Take out each part directly





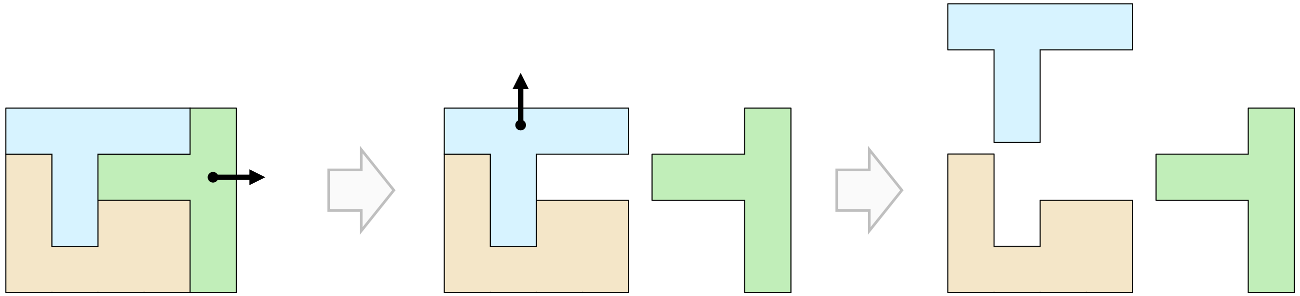
Disassembly plan



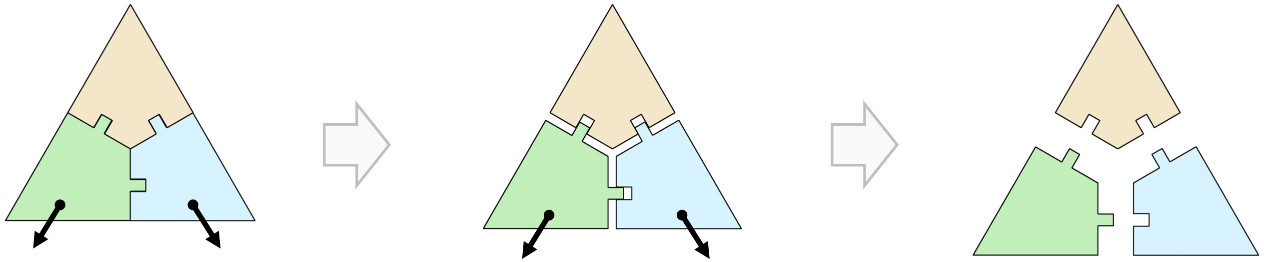
Classification of (Dis)assembly Plans

- Sequentiality
  - maximum number of moving sub-assemblies w.r.t one another in any (dis)assembly operation

sequential plan (**two**-handed plan)



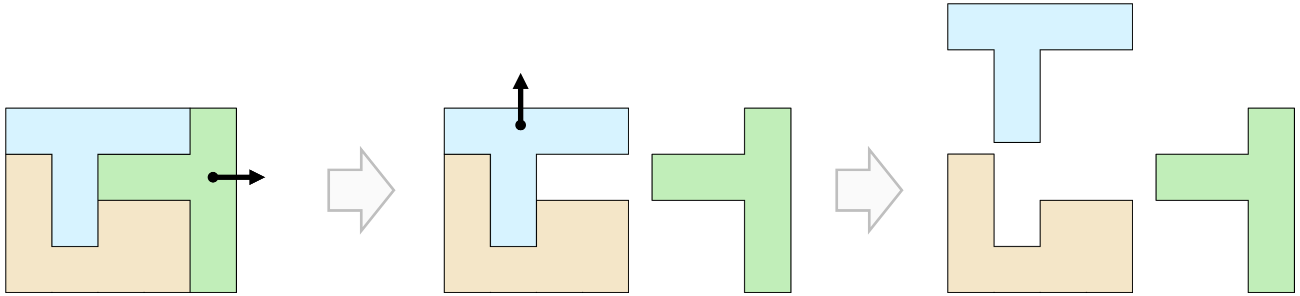
non-sequential plan (**three**-handed plan)



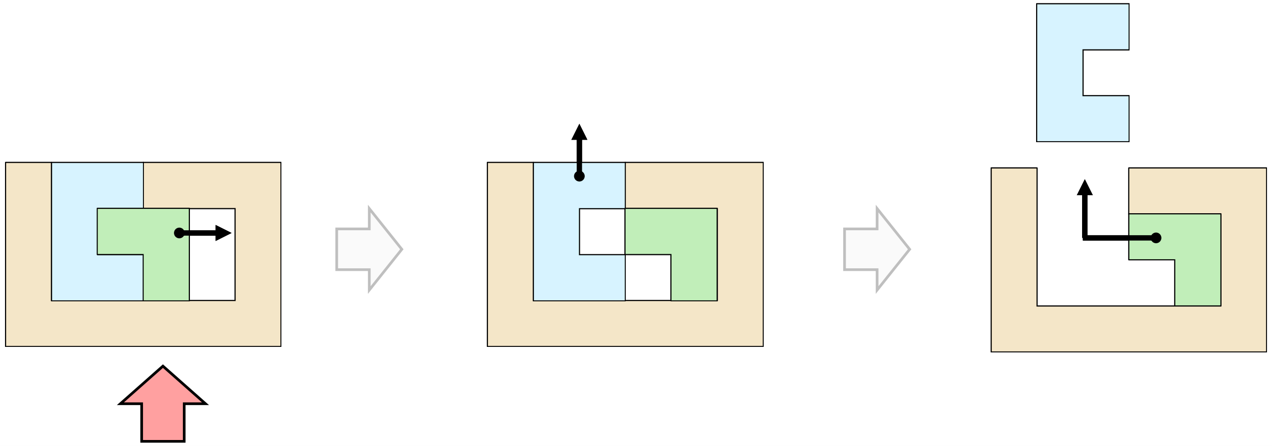
Classification of (Dis)assembly Plans

- Sequentiality
- Monotonicity
  - need for intermediate placement operations for at least one part of the assembly

monotone plan



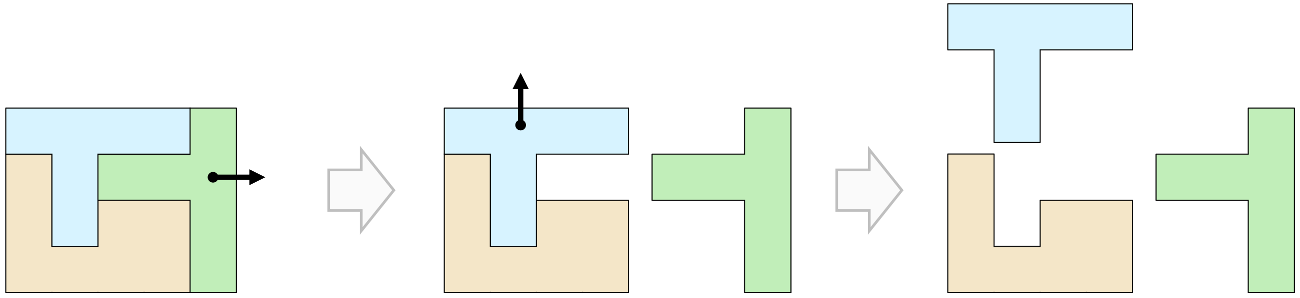
non-monotone plan



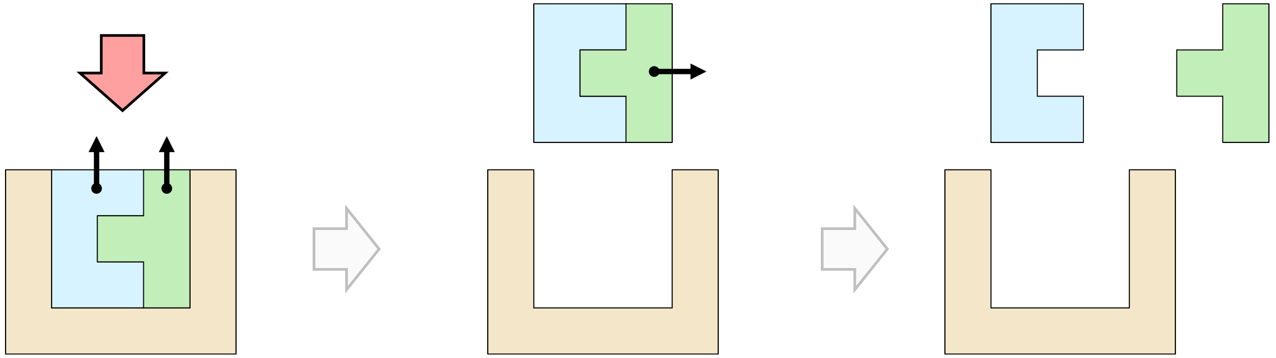
Classification of (Dis)assembly Plans

- Sequentiality
- Monotonicity
- Linearity
  - all assembly operations involve moving a single part with respect to the rest of the assembly

linear plan



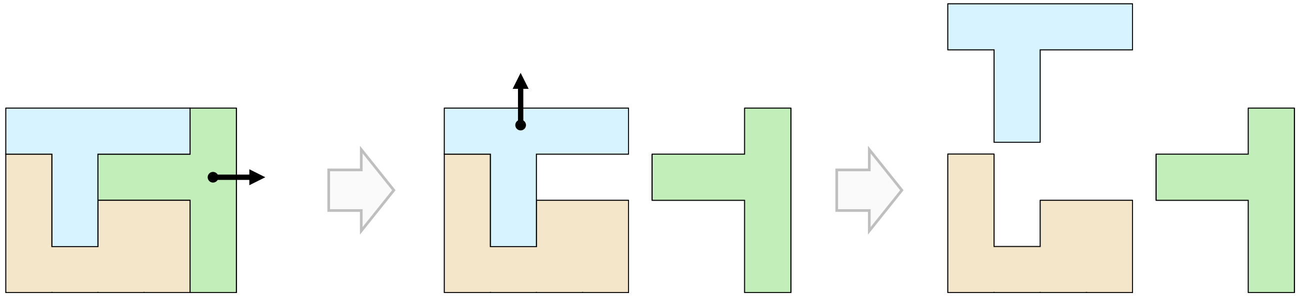
non-linear plan



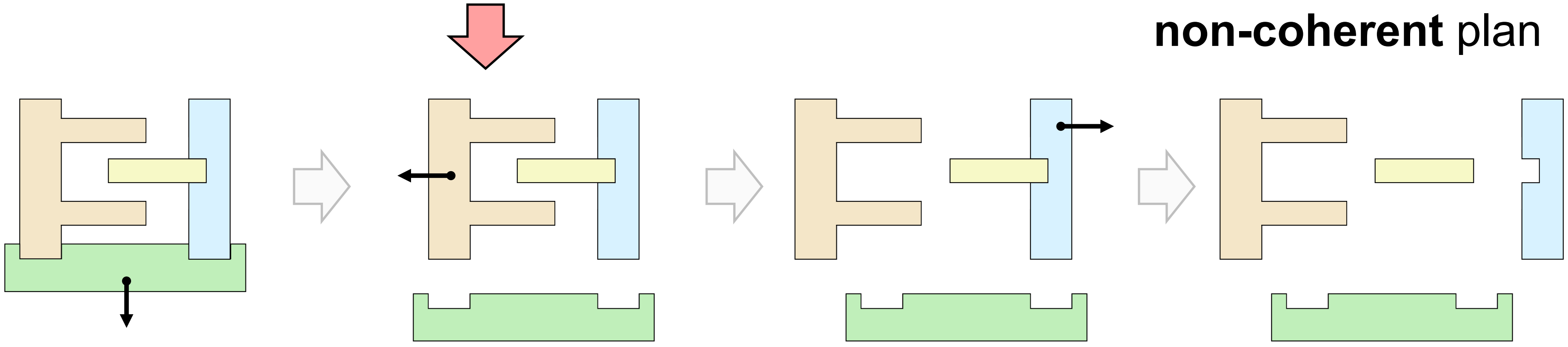
Classification of (Dis)assembly Plans

- Sequentiality
- Monotonicity
- Linearity
- Coherence
  - whether or not each part that is moved will touch the rest of the assembly

coherent plan

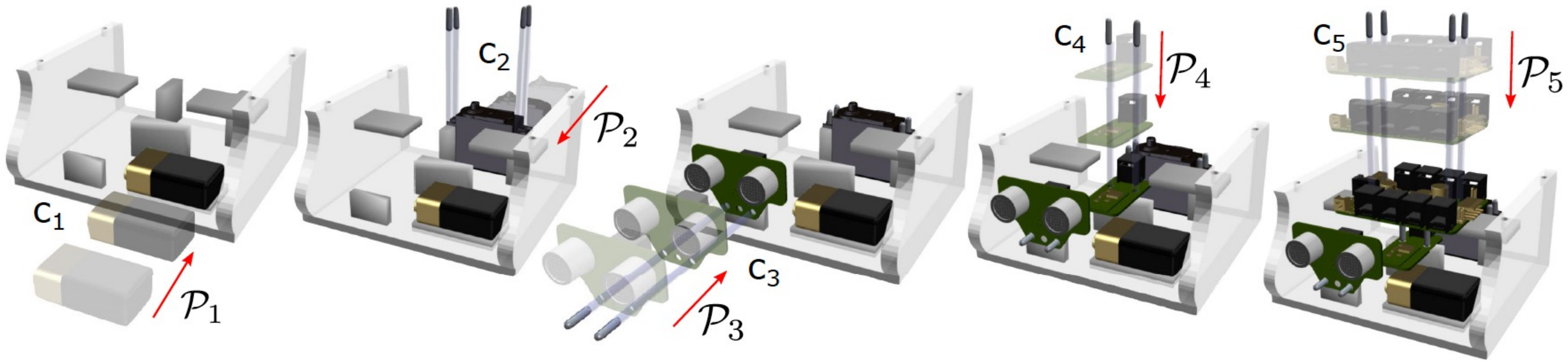


non-coherent plan



Complexity of (Dis)assembly Planning

- (Dis)assembly planning is an NP-complete problem
- The problem can be simplified by focusing on
  - sequential, monotone, linear, and coherent (dis)assembly plans
  - translational (dis)assembly motions

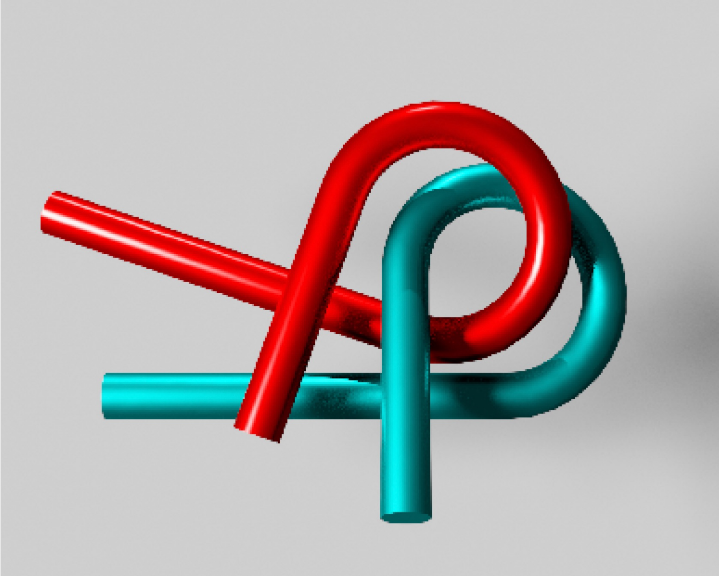


[Desai et al. 2018]

Research on Assembly Planning

- Study computational methods to search for complex (dis)assembly plans

non-coherent plan

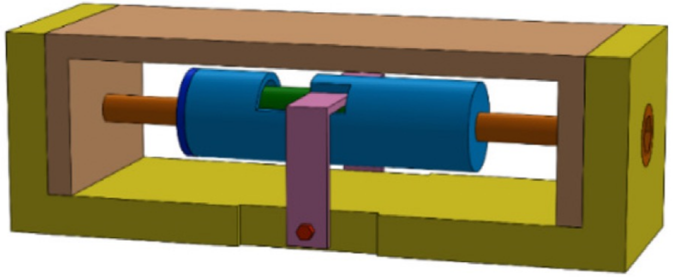


[Zhang et al. 2020]

non-sequential plan

[Araújo et al. 2019]

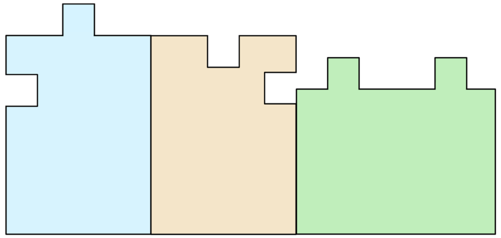
non-monotone plan



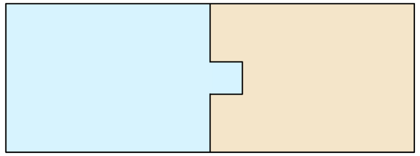
[Masehian et al. 2020]

Computational Analysis of Assemblies

Parts fabricability

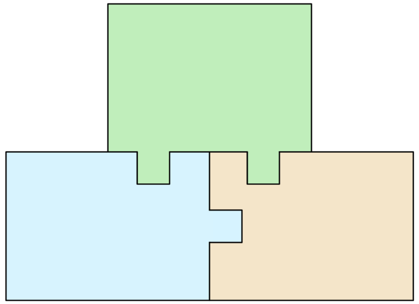


Parts joining



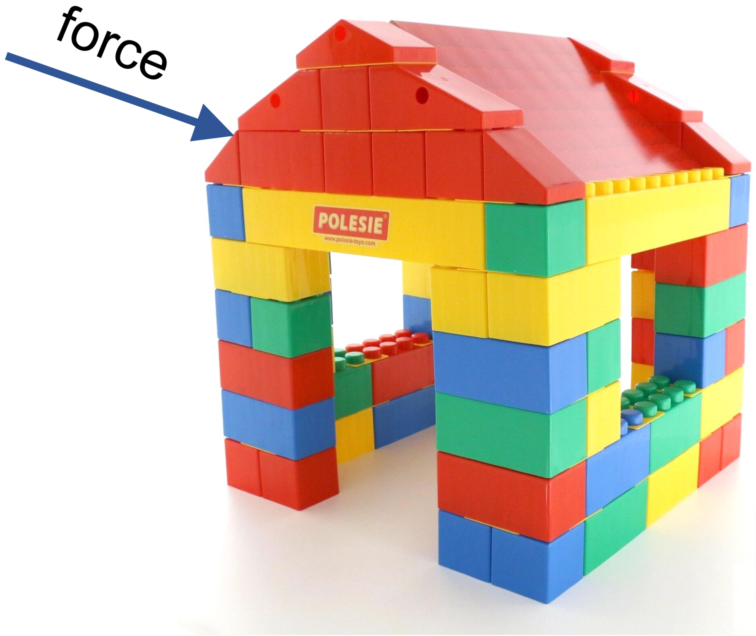
Assembly planning

Structural stability



Structural Stability

- An assembly with rigid parts is structurally stable if it can preserve its form under external forces without collapse
- How to define structural stability and analyze it computationally?



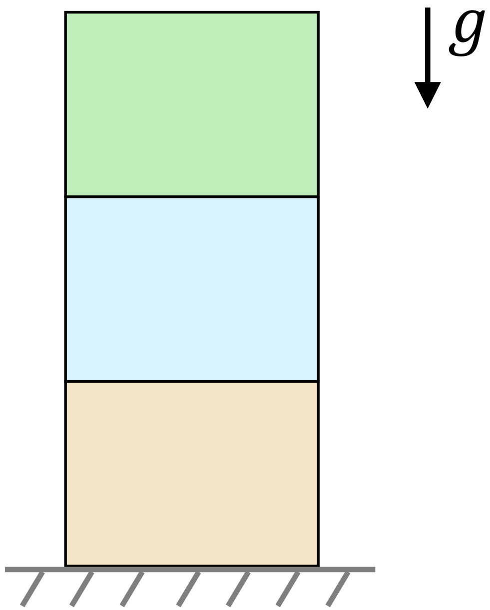
More stable!

ETH zürich imati
Peng Song Ziqi Wang Marco Livesu

Structural Stability

Static analysis

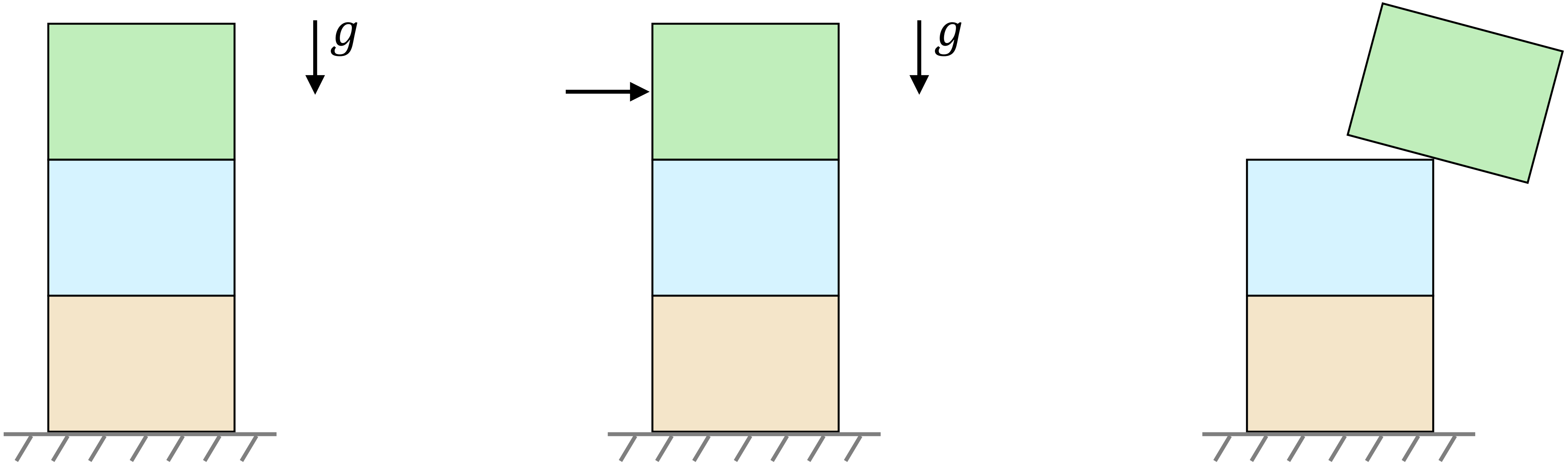
(equilibrium under
certain external forces)



Structural Stability

Static analysis

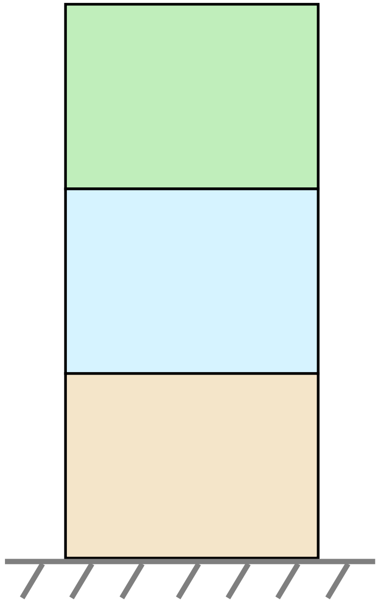
(equilibrium under
certain external forces)



Structural Stability

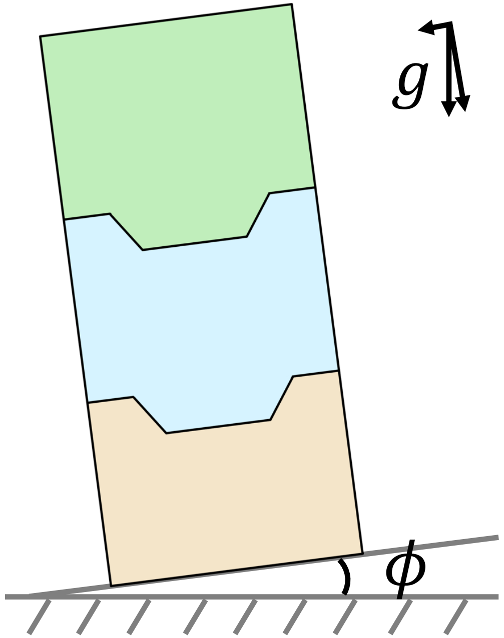
Static analysis

(equilibrium under
certain external forces)



Tilt analysis

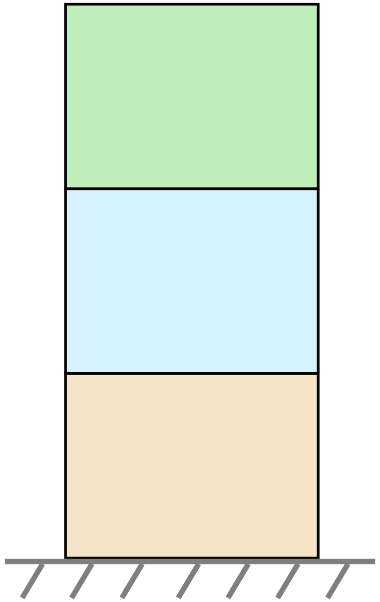
(equilibrium under
a set of lateral forces)



Structural Stability

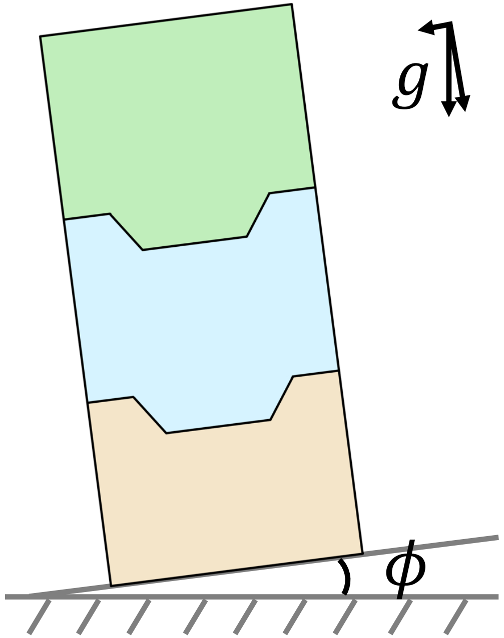
Static analysis

(equilibrium under
certain external forces)



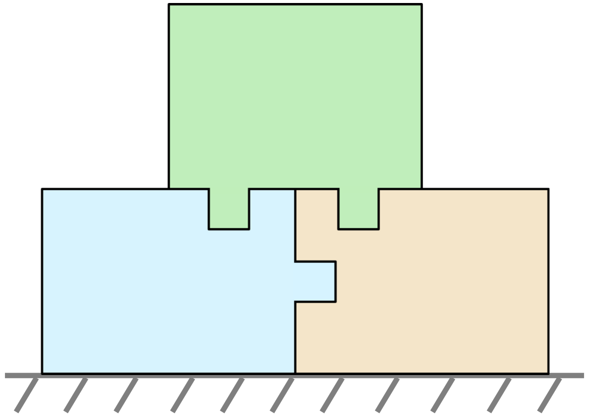
Tilt analysis

(equilibrium under
a set of lateral forces)



Interlocking test

(equilibrium under
arbitrary external forces)

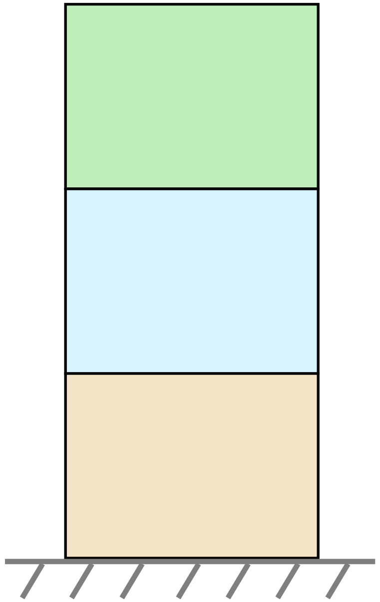


Peng Song Ziqi Wang Marco Livesu

Structural Stability

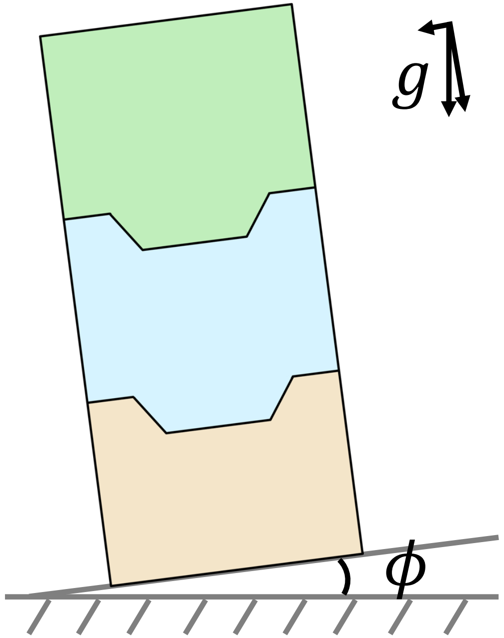
Static analysis

(equilibrium under
certain external forces)



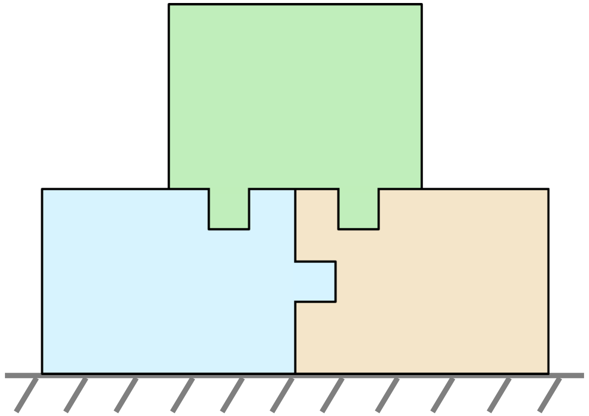
Tilt analysis

(equilibrium under
a set of lateral forces)



Interlocking test

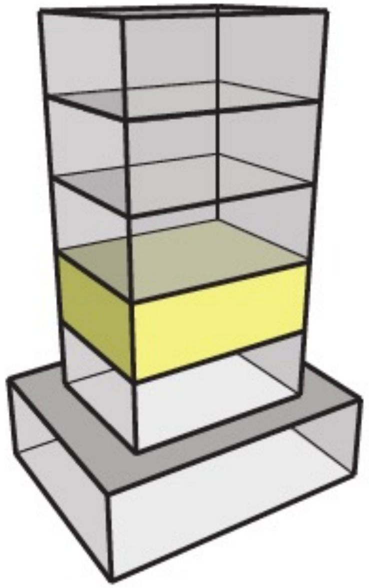
(equilibrium under
arbitrary external forces)

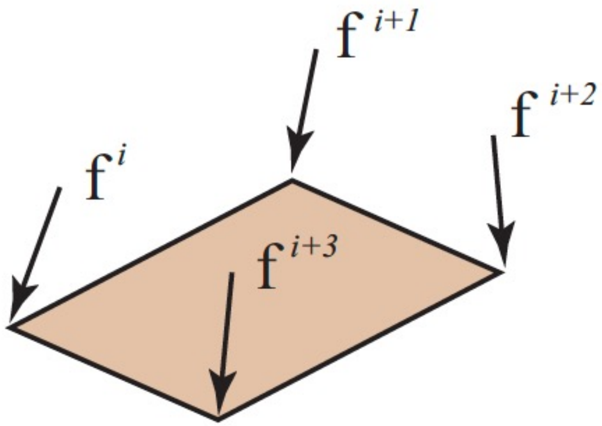


more structurally stable (with more restrictive joints)

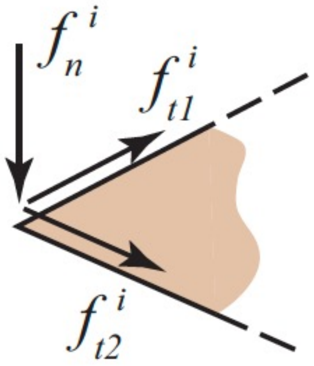
Static Analysis

- Rigid Block Equilibrium (RBE) method
  - A static equilibrium state means that there exists a network of interaction forces between the parts that can balance the external forces acting on each part





discretize interface forces

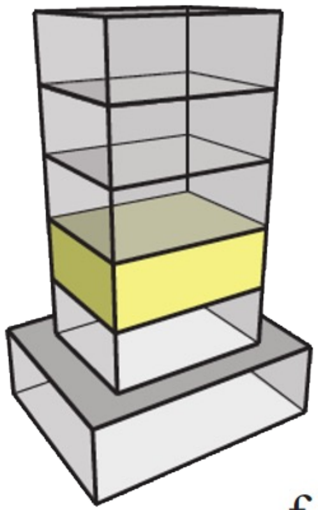


decompose interface force f^i

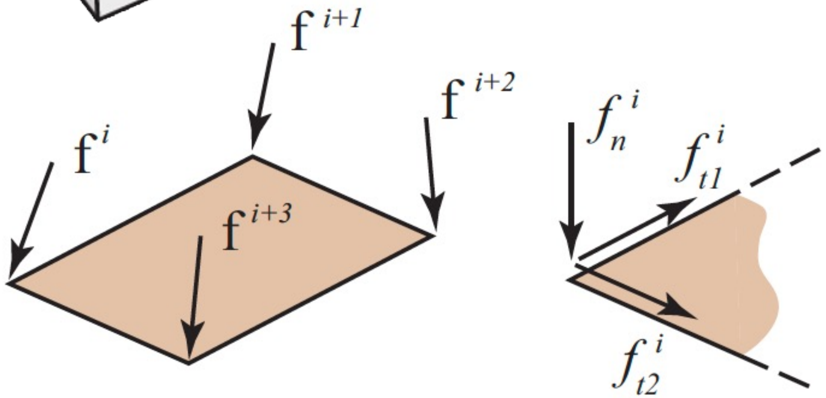
[Whiting et al. 2009]

Static Analysis

- Rigid Block Equilibrium (RBE) method



[Whiting et al. 2009]



Static Equilibrium

$$\mathbf{A}_{\mathbf{eq}} \cdot \mathbf{f} + \mathbf{w} = 0$$

matrix of equilibrium coefficients

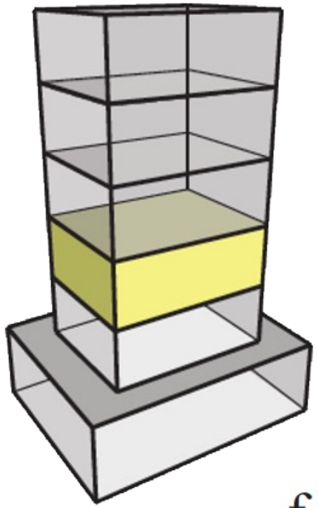
unknown vector of interface forces

external forces & torques

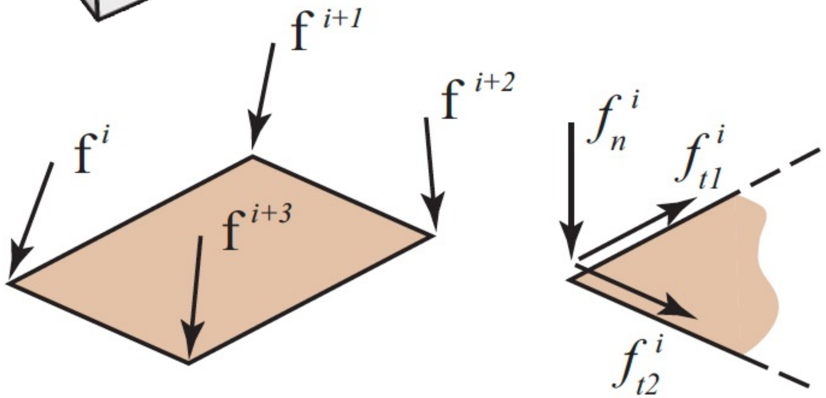
Static Analysis

- Rigid Block Equilibrium (RBE) method

a structure stands in **equilibrium** = an interface force solution **f exists**



[Whiting et al. 2009]



Static Equilibrium

$$\mathbf{A}_{\mathrm{eq}} \cdot \mathbf{f} + \mathbf{w} = 0$$

Compression Constraint

$$\mathrm{f}_\mathrm{n}^\mathrm{i} \geq 0, \quad \forall\ \mathrm{i} \in \text{ interface vertices}$$

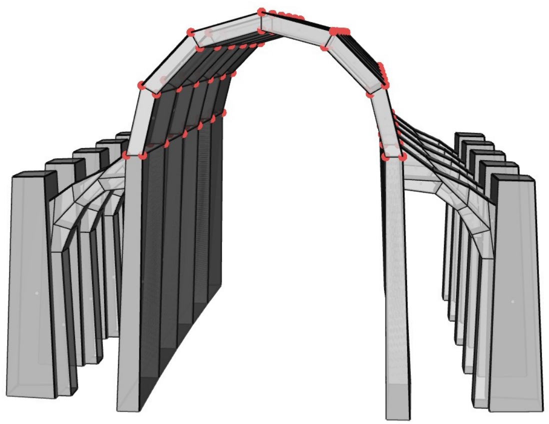
Friction Constraints

$$\left|f_{t1}^{i}\right|, \left|f_{t2}^{i}\right| \leq \alpha f_n^{i}, \quad \forall\ \mathrm{i} \in \text{ interface vertices}$$

Static Analysis

- RBE method and its improved versions have been used for analyzing and designing various rigid block structures that are in equilibrium

Masonry building

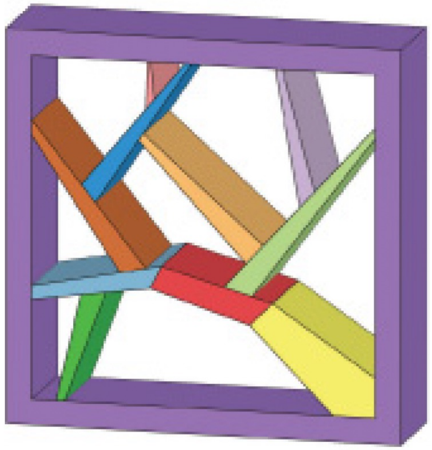


[Whiting et al. 2012]

Lego sculpture

[Luo et al. 2015]

Furniture



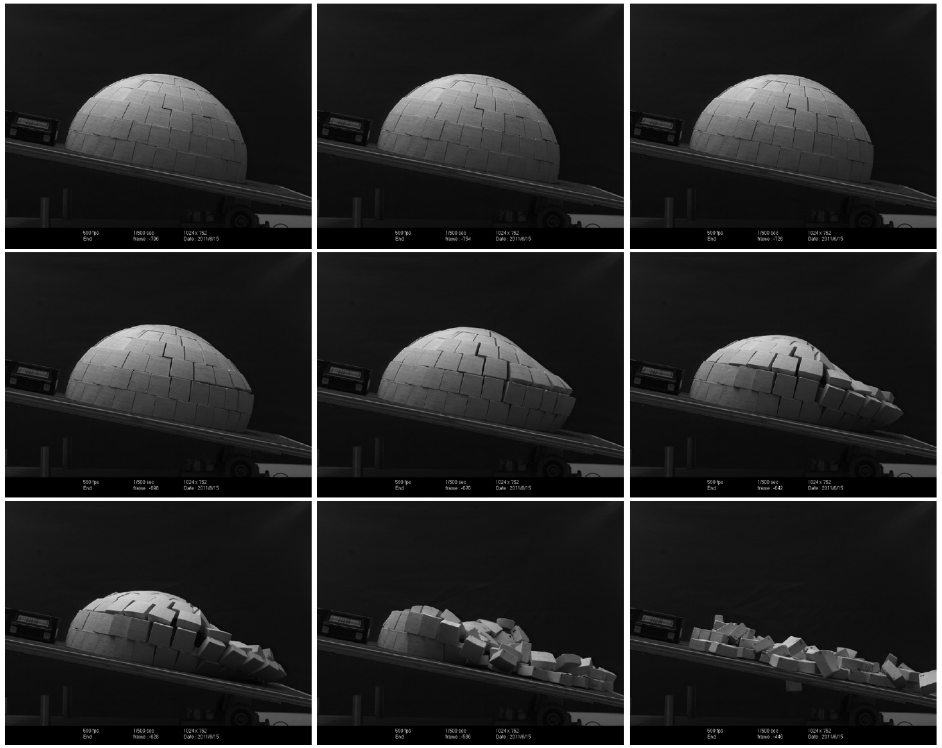
[Yao et al. 2017]

Equilibrium puzzle

[Wang et al. 2021]

Tilt Analysis

- **Rotate the ground plane** of an assembly to apply both a horizontal and vertical acceleration to the assembly
- The **critical tilt angle** $\boldsymbol{\phi}$ when the assembly collapse due to the lateral acceleration provides a measure of the structure's **lateral stability**

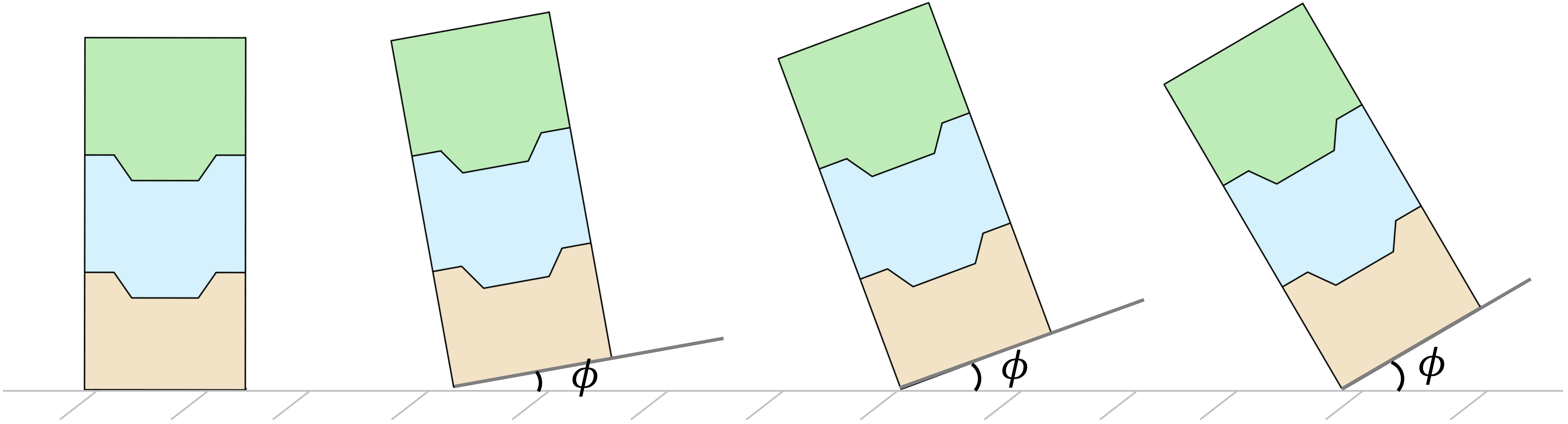


[Zessin 2012]

Tilt Analysis

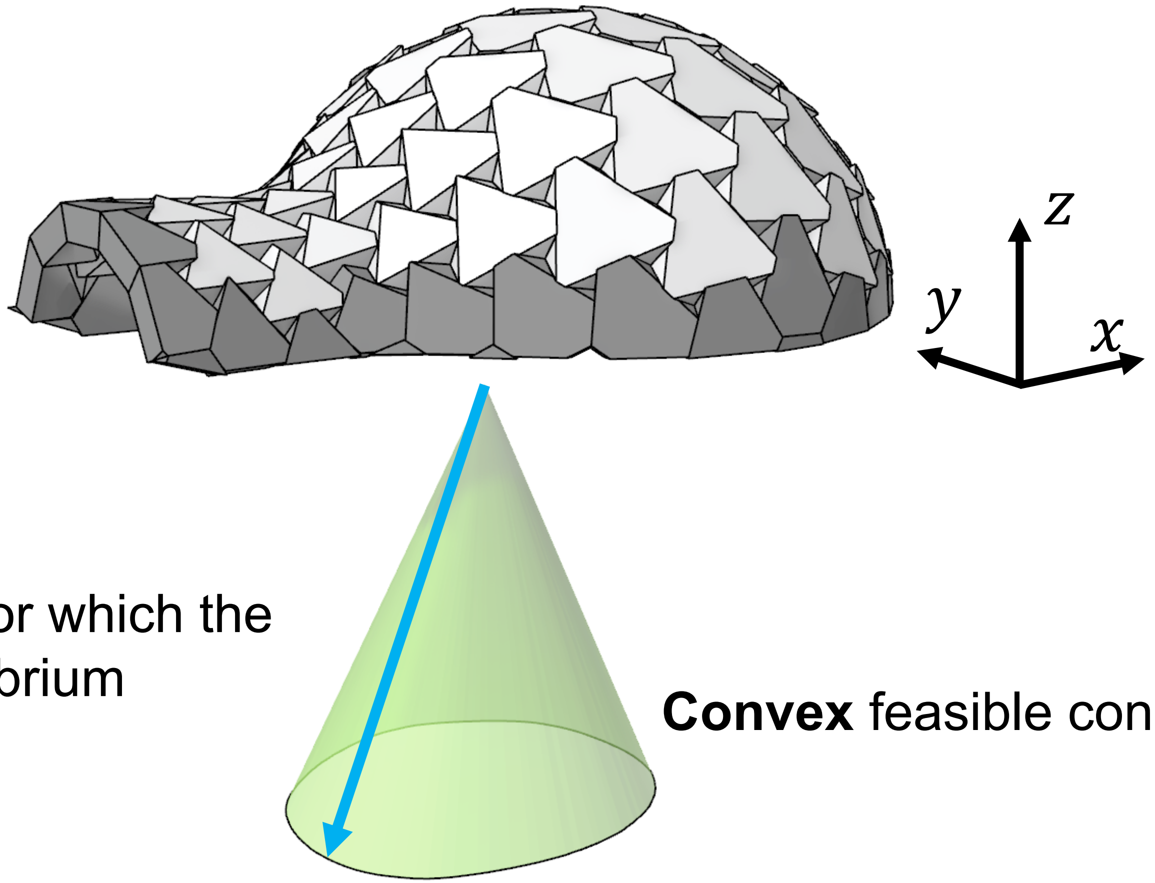
Stability Measure Φ = maximum tilt angle when the assembly is in equilibrium

- test whether the assembly is in equilibrium using the RBE method



Tilt Analysis

Stability Measure $\Phi = \min$ {maximum tilt angle for all tilt directions}



A gravity direction for which the structure is in equilibrium

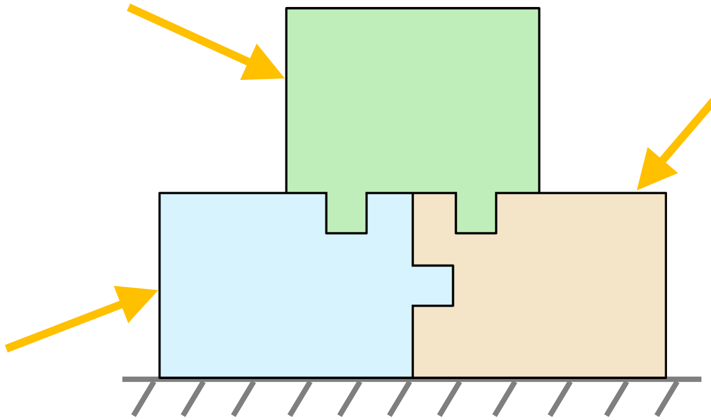
Convex feasible cone

[Wang et al. 2019]

Interlocking Test

Static

The assembly is in equilibrium under arbitrary external forces
(the key is held by other means)



Interlocking Test

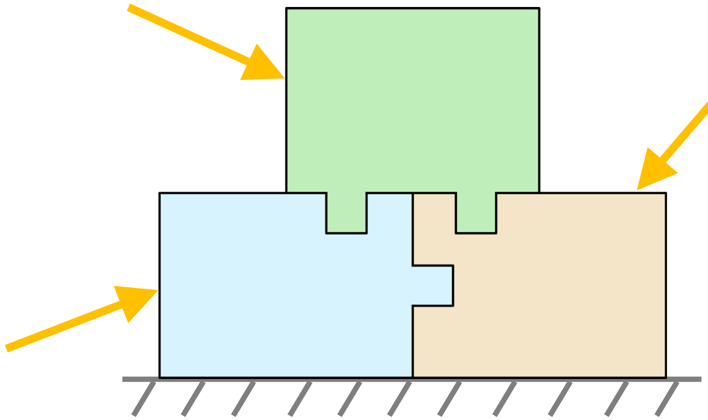
Dual

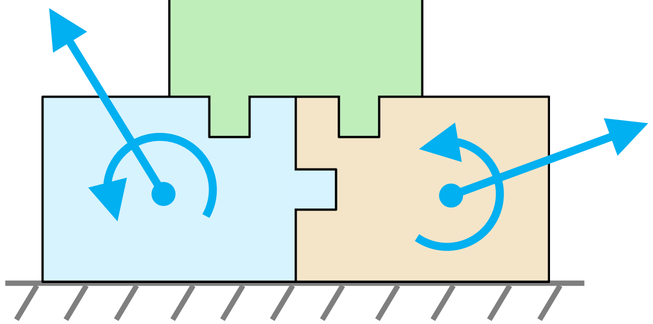
Static ⟷ **Kinematic**

The assembly is in equilibrium under arbitrary external forces
(the key is held by other means)

=

No movable part or subassembly
(except the key)

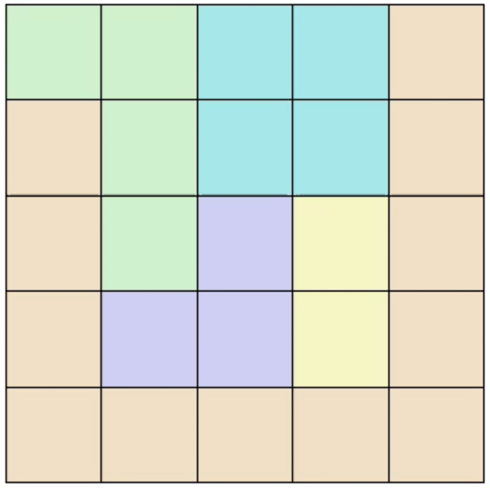




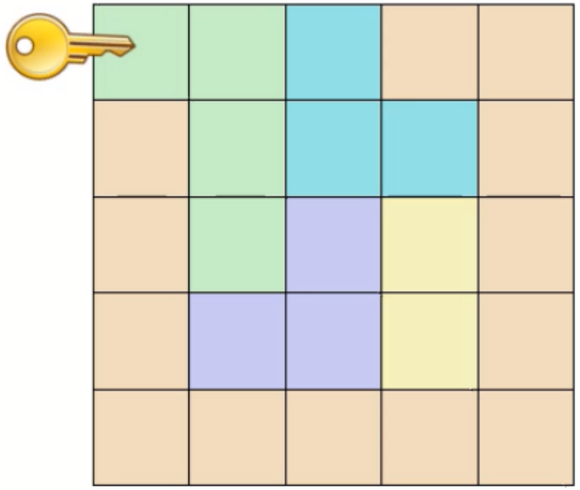
Peng Song Ziqi Wang Marco Livesu

Interlocking

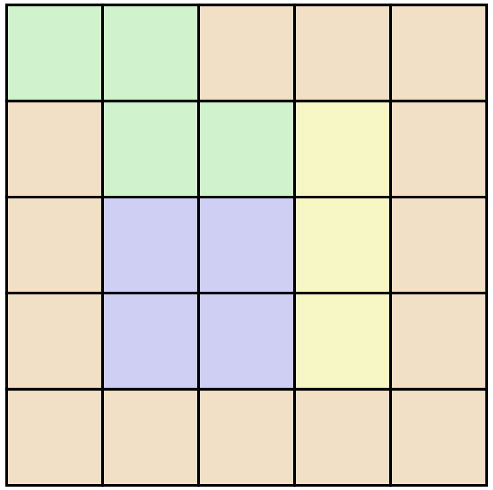
An assembly is interlocking if <u>only one movable part (key), while all other parts, as well as any subset of the parts, are immobilized</u>



non-interlocking



interlocking



deadlocking
(not disassemblable)

Interlocking

Interlocking assemblies have been designed for a variety of applications, where the assemblies need to bear forces from many different directions.

Puzzle

[Song et al. 2012]

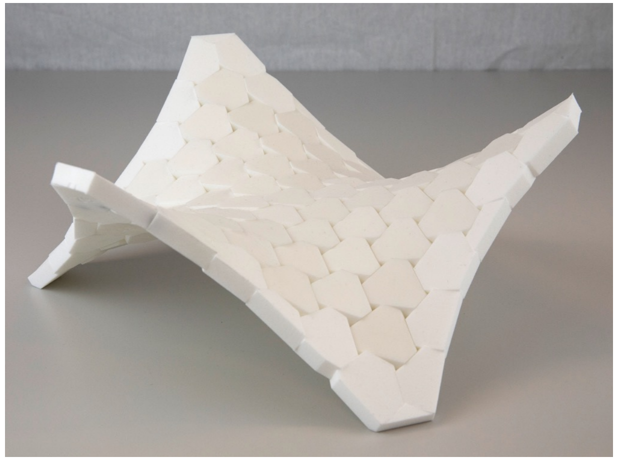
3D printing

[Yao et al. 2017]

Furniture

[Song et al. 2017]

Architecture

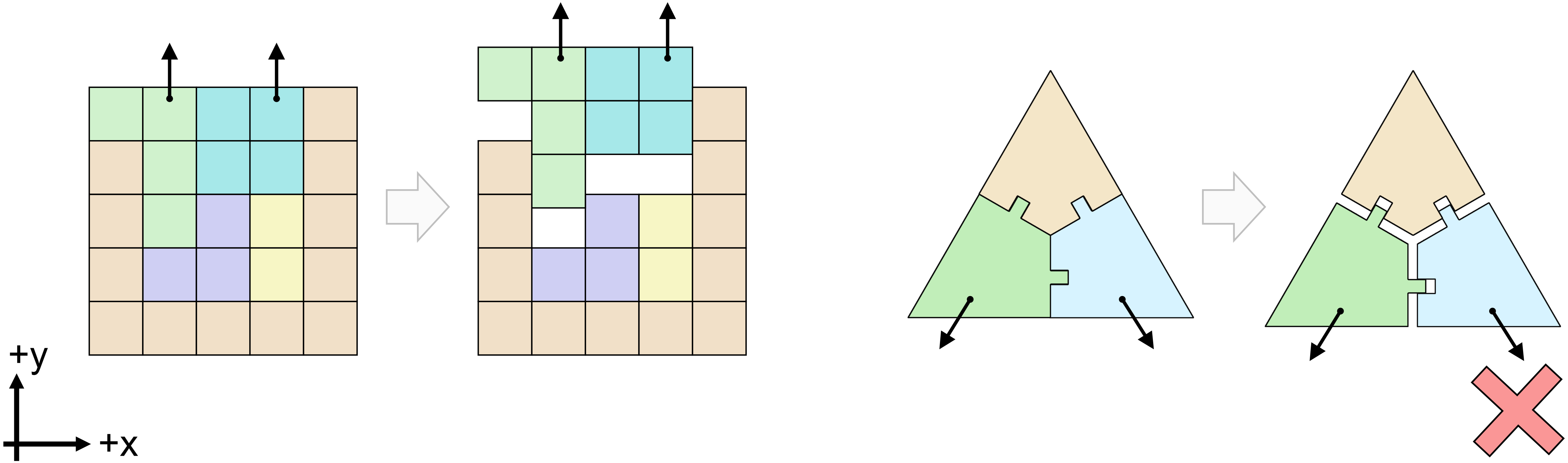


[Wang et al. 2019]

Interlocking Test

Method #1: check mobility of each part and each subset of the parts

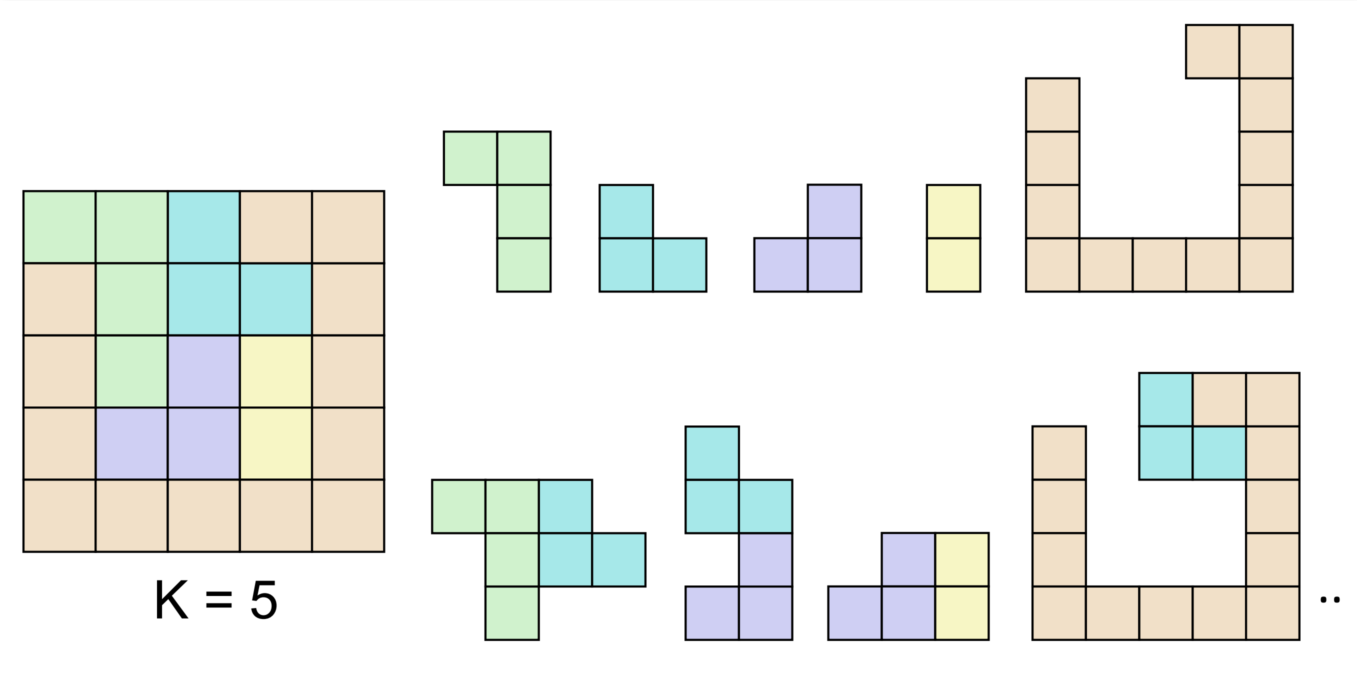
– assumption: parts in a subassembly always **translate** along the **same direction**



Interlocking Test

Method #1: check mobility of each part and each subset of the parts

- assumption: parts in a subassembly always **translate** along the **same direction**
- limitation: **exponential** computation complexity



[Song et al. 2012]

C(K, 1)

+

C(K, 2)

...

+

C(K, $\lfloor K/2 \rfloor$)

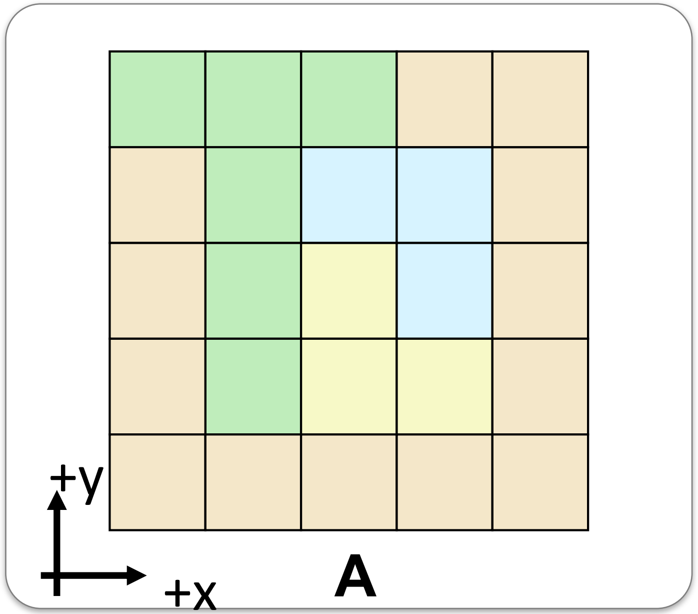
$= 2^{(K-1)}$ subsets of parts

Interlocking Test

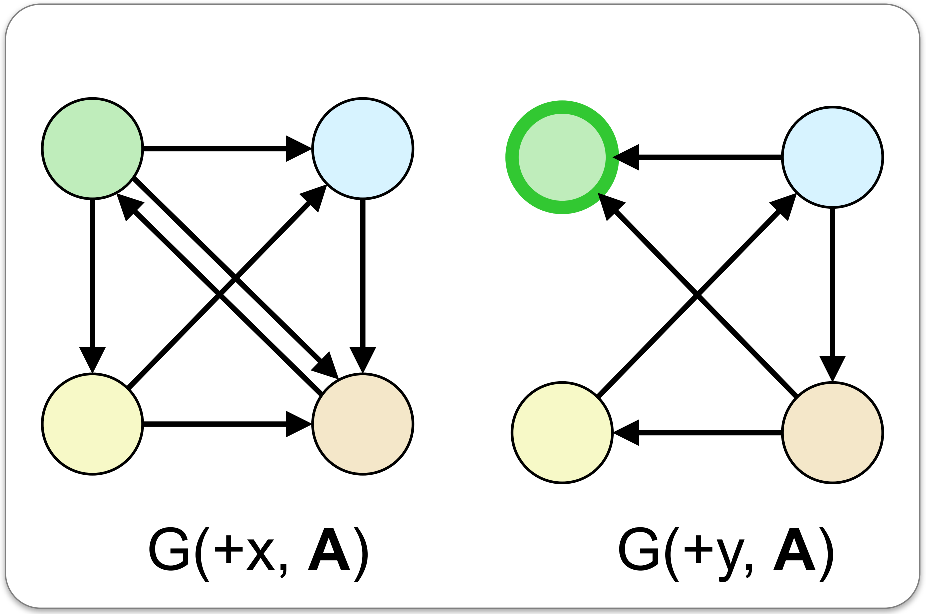
Method #2: check connectivity of direction blocking graphs (DBGs)

– assumption: parts in a subassembly always **translate** along the same **direction**

Test of global interlocking



Directional blocking graphs



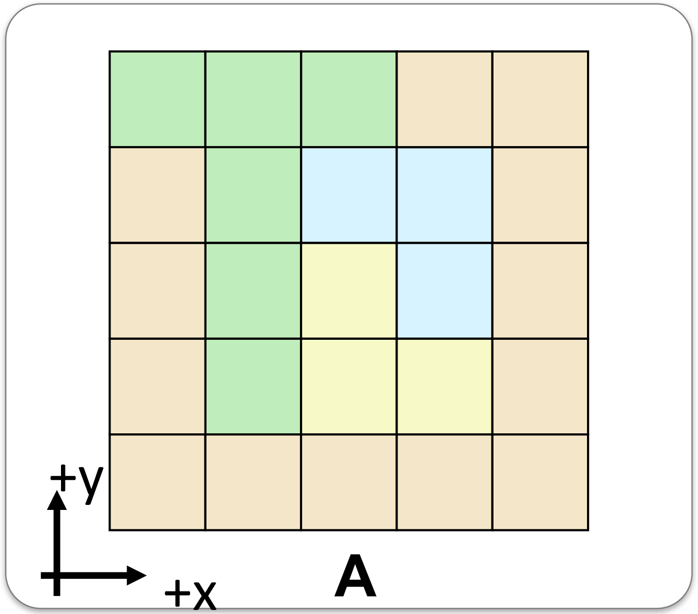
[Wang et al. 2018]

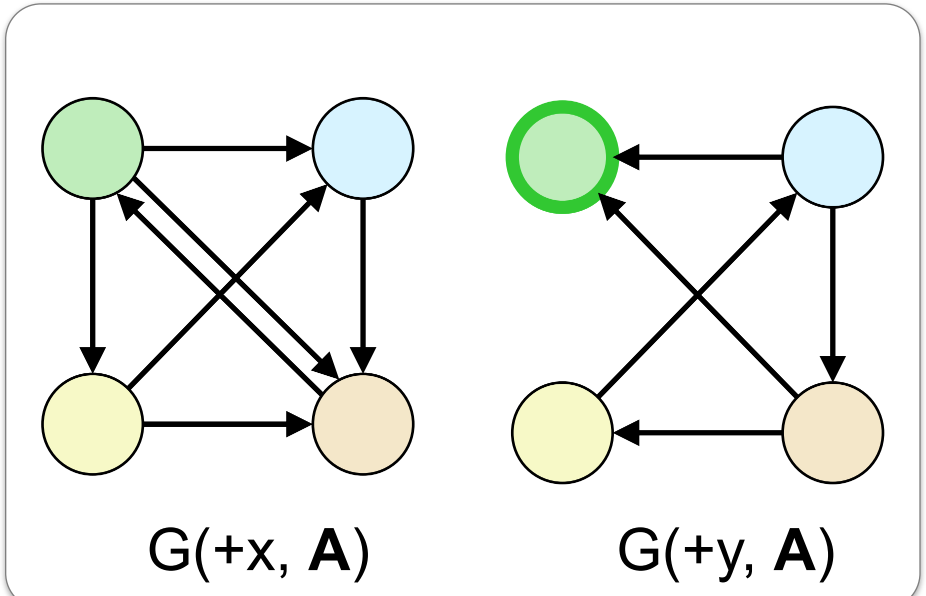
Interlocking Test

Method #2: check connectivity of direction blocking graphs (DBGs)

- assumption: parts in a subassembly always **translate** along the **same direction**
- advantage: **polynomial** time complexity

Assembly is interlocking = All DBGs are strongly connected (except the key part)



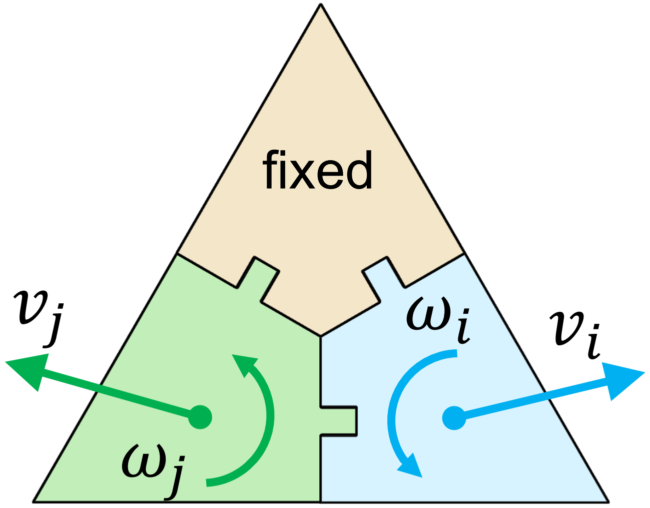


[Wang et al. 2018]

Interlocking Test

Method #3: inequality-based method

- **search space**: each part moves freely in the 3D space, with velocity $y_i = [v_i, \omega_i]$

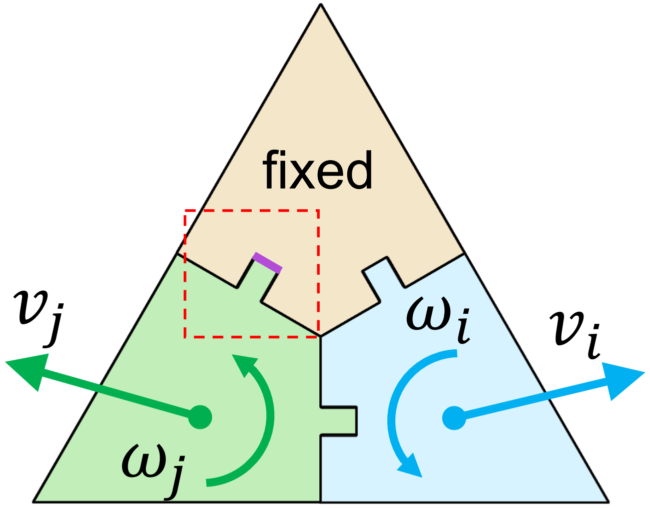


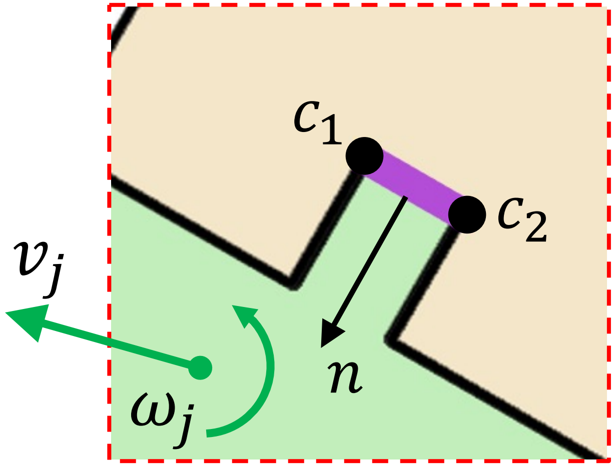
[Wang et al. 2019]

Interlocking Test

Method #3: inequality-based method

- **search space**: each part moves freely in the 3D space, with velocity $y_i = [v_i, \omega_i]$
- **constraint**: no collision between parts at each contact during infinitesimal movement





The contact constraints of a planar contact

$$v_{c1} \cdot n \geq 0$$

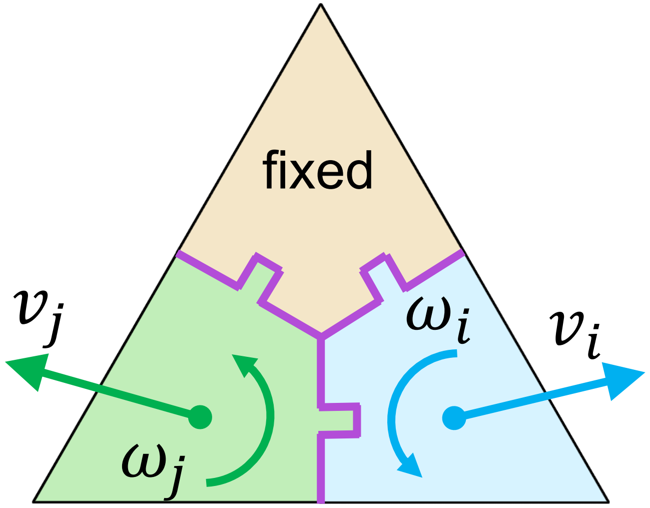
$$v_{c2} \cdot n \geq 0$$

[Wang et al. 2019]

Interlocking Test

Method #3: inequality-based method

- **search space**: each part moves freely in the 3D space, with velocity $y_i = [v_i, \omega_i]$
- **constraint**: no collision between parts at each contact during infinitesimal movement
- **formulation**: a system of linear inequalities by stacking the constraints



$$B\,\boldsymbol{Y} \geq \mathbf{0}$$

matrix of coefficients for collision-free constraints

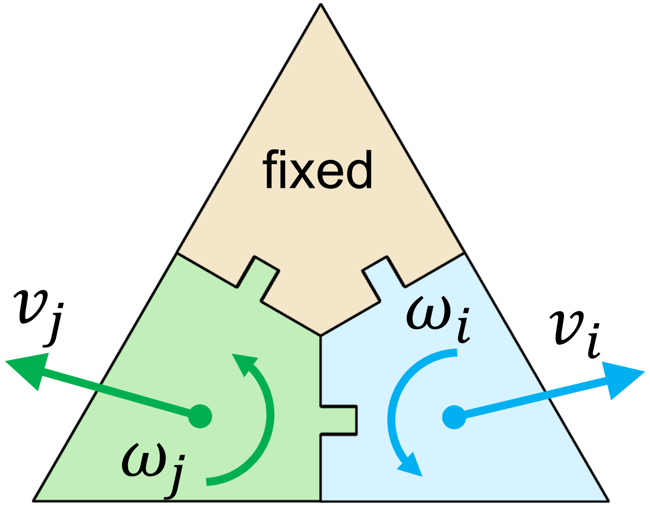
generalized velocity of the rigid body system

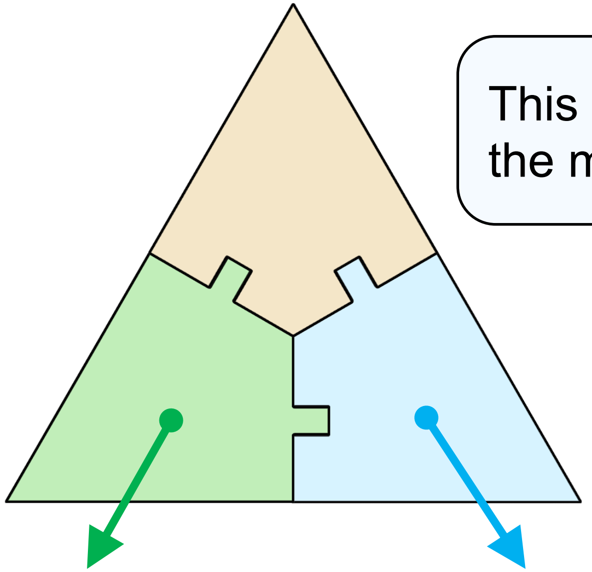
[Wang et al. 2019]

Interlocking Test

Method #3: inequality-based method

- **search space**: each part moves freely in the 3D space, with velocity $y_i = [v_i, \omega_i]$
- **constraint**: no collision between parts at each contact during infinitesimal movement
- **formulation**: a system of linear inequalities by stacking the constraints $B\,\boldsymbol{Y} \geq \mathbf{0}$
- **solve**: the assembly is interlocking if we cannot find such collision-free motion $\boldsymbol{Y}$





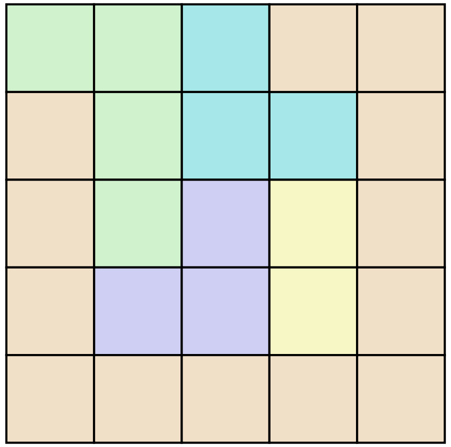
This assembly is not interlocking since the method can find collision-free motion

Interlocking Test

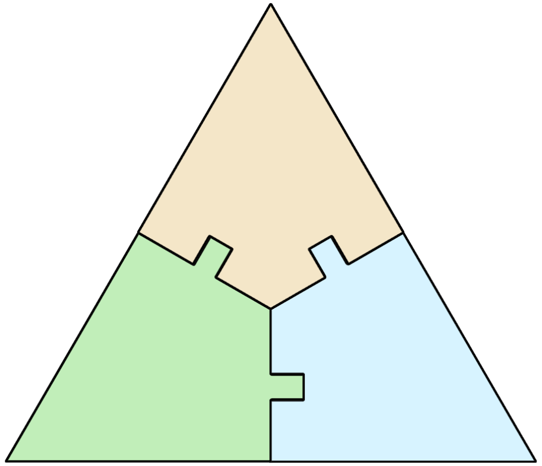
Method #3: inequality-based method

Advantage: a general interlocking test method

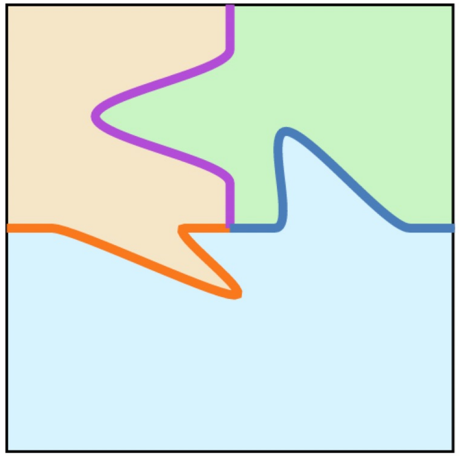
- parts can move along different directions simultaneously
- each part can translate and rotate



planar contacts
(axis-aligned)



single-direction joints



curved-contact joints

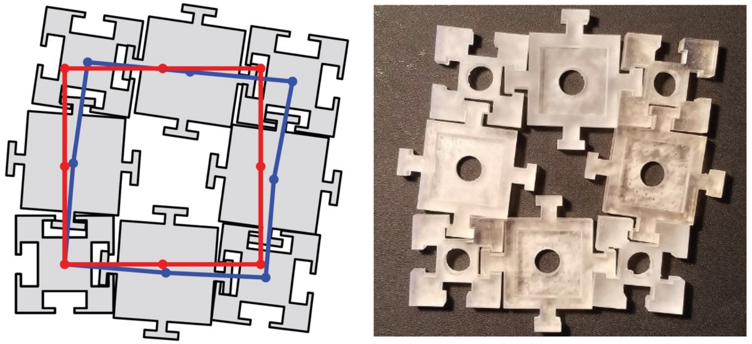
Research on Stability Analysis

- Stability analysis of assemblies with **friction**
- Stability analysis of assemblies under **tolerance**
- Stability analysis of assemblies with **deformable** parts

Coupled Rigid-Block Analysis

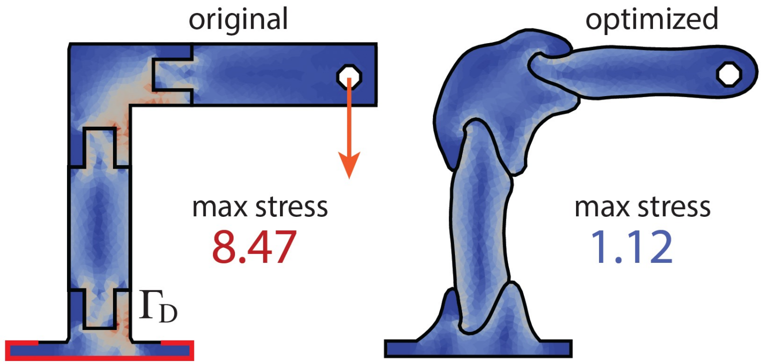
[Kao et al. 2022]

PuzzleFlex



[Lensgraf et al. 2020]

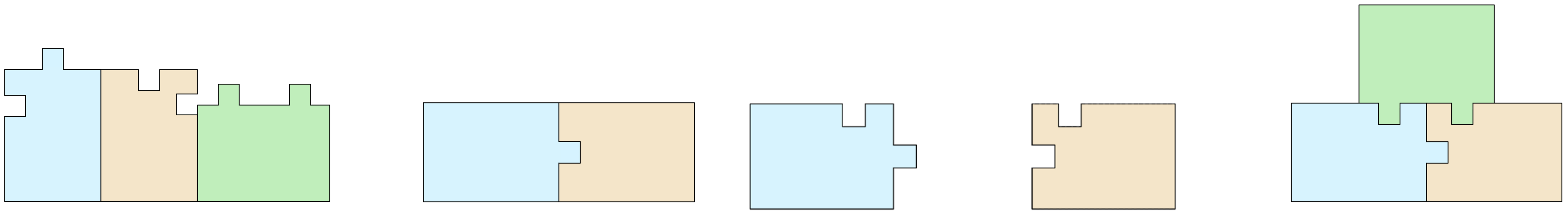
Contact-based Assemblies



[Tozoni et al. 2021]

Summary: Computational Analysis of Assemblies

- Introduce computational methods to analyze four aspects of assemblies
  - parts fabricability
  - parts joining
  - assembly planning
  - structural stability



Summary: Computational Analysis of Assemblies

- Introduce computational methods to analyze four aspects of assemblies
  - parts fabricability
  - parts joining
  - assembly planning
  - structural stability

- Future work: more aspects of assemblies for computational analysis
  - aesthetics
  - reconfigurability
  - functionality

  …

Timetable

		Peng	Ziqi	Marco
Introduction	~20 mins	X		
Computational analysis of assemblies	~50 mins	X		
Computational design of assemblies	~50 mins		X	
Computational fabrication of assemblies	~50 mins			X
Q & A	~10 mins	X	X	X